RADIO FREQUENCY FINGERPRINTING TECHNIQUES THROUGH PREAMBLE MODIFICATION IN IEEE 802.11B

THESIS

Nicholas J. Kulesza, Captain, USAF

AFIT–ENG–T–14–J–8

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT–ENG–T–14–J–8

RADIO FREQUENCY FINGERPRINTING TECHNIQUES THROUGH PREAMBLE MODIFICATION IN IEEE 802.11B

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Computer Science

Nicholas J. Kulesza, B.S.C.S.

Captain, USAF

June 2014

AFIT–ENG–T–14–J–8

RADIO FREQUENCY FINGERPRINTING TECHNIQUES THROUGH PREAMBLE MODIFICATION IN IEEE 802.11B

Nicholas J. Kulesza, B.S.C.S.
Captain, USAF

Approved:

/signed/	28 May 2014
Barry E. Mullins, PhD (Chairman)	Date
/signed/	28 May 2014
Timothy H. Lacey, PhD (Member)	Date
/signed/	28 May 2014
Michael A. Temple, PhD (Member)	Date

Abstract

Wireless local area networks are particularly vulnerable to cyber attacks due to their contested transmission medium. Access point spoofing, route poisoning, and cryptographic attacks are some of the many mature threats faced by wireless networks. Recent work investigates physical-layer features such as received signal strength or radio frequency fingerprinting to identify and localize malicious devices. This thesis demonstrates a novel and complementary approach to exploiting physical-layer differences among wireless devices that is more energy efficient and invariant with respect to the environment than traditional fingerprinting techniques. Specifically, this methodology exploits subtle design differences among different transceiver hardware types.

A software defined radio captures packets with standard-length IEEE 802.11b preambles, manipulates the recorded preambles by shortening their length, then replays the altered packets toward the transceivers under test. Wireless transceivers vary in their ability to receive packets with preambles shorter than the standard. By analyzing differences in packet reception with respect to preamble length, this methodology distinguishes amongst eight transceiver types from three manufacturers. All tests to successfully enumerate the transceivers achieve accuracy rates greater than 99%, while transmitting less than 60 test packets.

This research extends previous work illustrating RF fingerprinting techniques through IEEE 802.15.4 wireless protocols. The results demonstrate that preamble manipulation is effective for multi-factor device authentication, network intrusion detection, and remote transceiver type fingerprinting in IEEE 802.11b.

To God.

Acknowledgments

I would like to take the time to acknowledge several individuals who were instrumental to the production of this thesis. First, I am very fortunate to have the support of Dr. Barry Mullins as an academic advisor. His feedback and guidance were invaluable throughout this process.

I owe a tremendous amount of credit to Captain Ben Ramsey, who lended an open ear and answered many of my questions in areas I lacked experience in. I would also like to personally thank Captain Soloman Sonya for giving me an opportunity to accompany him, presenting at various conferences.

I would also like to thank my parents for continuing to push me towards success. Lastly, I can not thank enough my wonderful wife for her support throughout this journey. This thesis would not have been possible without her encouragement and support.

Nicholas J. Kulesza

Table of Contents

Page

Page

List of Figures

Figure Page

List of Tables

List of Acronyms

Acronym	Definition
AP	Access Point
ARP	Address Resolution Protocol
CCA	Clear Channel Assessment
CCK	Complementary Code Keying
CDF	Cumulative Distribution Function
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
dB	Decibels
DBPSK	Dynamic Binary Phase Shift Keying
DNS	Domain Name Service
DQPSK	Dynamic Quadrature Phase Shift Keying
DSL	Digital Subscriber Lines
DSS	Distribution System Services
DSSS	Direct Sequence Spread Spectrum
ESS	Extended Service Set
ETSI	European Telecommunications Standard Institute
FCS	Frame Check Sequence
FHSS	Frequency Hopping Spread Spectrum
GHz	Gigahertz
Gbit	Gigabit
HIPERLAN	High Performance Local Area Network
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers

Acronym	Definition
IP	Internet Protocol
ISM	Industrial, Scientific and Medical
LLC	Logic Link Control
MAC	Media Access Control
Mbps	Mega–bits per second
MITM	Man–in–the–Middle
MPDU	MAC Protocol Data Unit
MSDU	MAC Service Data Unit
Msps	Million samples per second
OFDM	Orthogonal Frequency Division Multiplexing
OSI	Open Systems Interconnection
PAN	Personal Area Network
PBCC	Packet Binary Convolution Coding
PHY	Physical Layer
PLCP	Physical Layer Convergence Procedure
PMD	Physical Medium Dependent
PPDU	Physical Layer Protocol Data Unit
PSDU	Physical Layer Service Data Unit
RF	Radio Frequency
RSSI	Received Signal Strength Indicator
SFD	Start Frame Delimiter
SNRA	Sequence Number Rate Analysis
STA	Station
SXS	Signals eXploitation System
USB	Universal Serial Bus

Acronym	Definition
USRP	Universal Software Radio Peripheral
WLAN	Wireless Local Area Network

RADIO FREQUENCY FINGERPRINTING TECHNIQUES
THROUGH PREAMBLE MODIFICATION IN IEEE 802.11B

I. Introduction

1.1 Motivation

Wireless technologies represent a field of continual evolution, providing Internet access to homes and businesses. This widespread propagation acts as the catalyst of the digital information age. Electronic commerce represents a multi-billion dollar industry. Social media connects millions across the planet. The Internet opens a new domain usable by anyone with access to a computer with Ethernet, fiber optic or other wireless capabilities.

The explosive growth of wireless technologies that provide access to the Internet presents a series of unique security challenges. Several developments in sound security practices address these issues. Attackers exploit a feature in the data link layer of the Open Systems Interconnection (OSI) model, which permits arbitrary Media Access Control (MAC) assignment in a wireless transceiver. The abuse of unique digital identifiers, known as "MAC spoofing", leads to potential malicious activity. One of the most significant challenges today addresses how to thwart would-be attackers from abusing arbitrary MAC assignment.

There exist creative solutions that take advantage of independent manufacturing, with respect to wireless transceivers. This thesis explores the potential to enumerate hardware devices that provide wireless connectivity to the Internet. Successful device classification suggests a deterministic approach that compares transceiver performance with the expected outcome for the corresponding MAC address. Theoretically, the solution to combat MAC spoofing is to simply validate the MAC address.

1.2 Hypothesis

If transceivers differ in their response to a series of packets containing modified preambles, the packet response rates characterizing each transceiver suggests the possibility of device classification. Prior to handling received wireless traffic, transceivers listen for a preamble to perform a series of network synchronization functions. If the window to perform these functions diminishes, there exists the possibility that network synchronization fails and the transceiver drops the altered packet. Institute of Electrical and Electronics Engineers (IEEE) defines standards applicable to the IEEE 802.11 protocol, however the standard does not reach the level of specificity to dictate physical layer implementation. This research hypothesizes that if wireless transceivers respond differently upon receipt of a packet containing a modified preamble, RF fingerprinting is possible.

1.3 Research Goals

The overall goal of this research is to perform device classification over IEEE 802.11b (hereafter referred to as "802.11b"), utilizing an RF fingerprinting technique. This research analyzes behavioral differences in various wireless transceivers to discern the prospect of device classification. In addition, the research also seeks to accurately determine transceiver types efficiently by minimizing the required number of packets to successfully enumerate a transceiver. In order to achieve these goals, multiple experiments are conducted to generate a transceiver profile and gauge the probability of packet response to manipulated preambles.

1.4 Scope of Research

Pursuant to the goals of this research, there exist a series of assumptions that bind the scope of this experiment. The collective pool of eight transceivers represent three large scale manufacturers: Atheros, Broadcom and Intel. Other manufacturers that produce wireless transceivers are unobtainable through cost–effective means. Packet response

rates during the experiment extend exclusively to 802.11b. The tests utilize standard long preambles within 802.11b, as opposed to the optional shorter preamble. Network conditions include four to six other wireless access points operating between 2.412 - 2.462 Gigahertz (GHz). Preamble modifications range from a single bit up to ten bits and packet replays occur through the National Instruments Ettus Universal Software Radio Peripheral (USRP).

1.5 Execution

Each transceiver transmits a series of Internet Control Message Protocol (ICMP) echo requests to a wireless access point. The USRP captures the RF signals and creates a binary file, which contains the instantaneous amplitude readings for the RF signals. MATLAB software provides a graphical user interface to plot the data, identify the region of interest and alter the specific samples to create the trial test packets. The experiment comprises of ten sets of packets, each set removes another bit from the preamble. Once created, the USRP retransmits the modified RF signals back to the transceiver. Wireshark monitors network traffic on the transceiver under test to account for positive and negative responses to each transmission. Section 3.2.2 covers the detailed approach concerning this research.

1.6 Thesis Layout

This chapter addresses the motivation, hypothesis, assumptions and goals of this research. This chapter also briefly describes the experimental design. Chapter 2 covers the literature review and also highlights related research efforts that parallel this methodology, summarizing their results. Chapter 3 discusses the experimental design of this research and illustrates how the experiments are executed. Chapter 4 provides statistical analysis of the experimental results. Chapter 5 reports the conclusions of the findings, and identifies where future research can expand the prospect of device classification through RF fingerprinting techniques.

II. Background and Literature Review

2.1 Introduction

This chapter addresses several topics pertinent to RF fingerprinting. Section 2.2 provides an introduction to the 802.11 protocol specification and defines unique characteristics of specific protocols. Section 2.3 defines several specifications that hardware devices must fulfill to meet the standards of 802.11b. Section 2.4 illustrates frame classes and relationships between states. Section 2.5 details the relationship between the data link and physical layers, as headers and trailers are appended to wireless network traffic. In addition, Section 2.5 also details preambles in 802.11, network synchronization and channel estimation. Section 2.6 introduces the technical specifications of the National Instruments USRP. Section 2.7 addresses security concerns stemming from arbitrary MAC address assignment. Section 2.8 describes prior existing techniques to validate MAC addresses and also analyzes potential technical solutions within preamble modifications.

2.2 Advent of Wireless Technologies

During the 1990s, commercial and residential networks increasingly connected to the Internet. Telephone lines failed to provide the necessary bandwidth for users to access complex web-pages or download multimedia from the Internet [KRT02]. This resulted in lag and lengthy delays, while the requested resources buffered through the line strained at the maximum throughput rate of 56 Kbps [KuR10]. Consumer demand transitioned to a more favorable Internet connection with cable and Digital Subscriber Lines (DSL), which offered 15 times greater bandwidth for information to flow. At the same time, laptop computing became popular in the office environment for their portability.

Mobile computing ignited the demand for mobile connectivity, an ability to operate free of cables and utilize Radio Frequency (RF) to send and receive data. Initially, two

separate standards were developed. The European Telecommunications Standard Institute (ETSI) developed the High Performance Local Area Network (HIPERLAN) [DAB+02]. Meanwhile, the IEEE developed the currently–accepted standards of the 802.11 families, which included the Wireless Local Area Network (WLAN). The 802.11 standard gained widespread acceptance and subsequently resulted in the demise of HIPERLAN.

As shown in Table 2.1, IEEE 802.11 operates exclusively on the 2.4 GHz band [Voc12]. Direct Sequence Spread Spectrum (DSSS) and Frequency Hopping Spread Spectrum (FHSS) provide two modulation techniques, enabling the transmission rates of 1 or 2 Mbps. In September 1999, 802.11a was ratified, dramatically increasing the available throughput of the original 802.11 standard. 802.11a utilized Orthogonal Frequency Division Multiplexing (OFDM). Operating at 5 GHz as opposed to the traditional 2.4 GHz range mitigates packet loss and collisions, since more hosts and access points still utilize the 2.4 GHz frequency. IEEE ratified the 802.11b standard in September 1999, around the same time as 802.11a. Despite a maximum throughput of 11 Mbps, as opposed to 54 Mbps in 802.11a, the 802.11b standard performs better at lower speeds and does not lose signal strength through absorption as easily [KuR10].

Table 2.1: 802.11, 802.11a and 802.11b Specifications

Protocol	Frequency	Transmission Rates	Modulation Technique	Range
802.11	2.4 GHz	1, 2 Mbps	DSSS, FHSS	65 ft
802.11a	5 GHz	6, 9, 12, 18, 24, 36, 48, 54 Mbps	OFDM	115 ft
802.11b	2.4 GHz	1, 2, 5.5, 11 Mbps	DSSS	115 ft

2.3 IEEE 802.11 Distribution System Services

All access points and their respective clients that transmit data in the wireless medium must provide services in accordance to IEEE 802.11. Clients that utilize 802.11 facilitate services, including: association, disassociation, distribution, integration and re-

association. Critical to this research, IEEE does not specify how a message is sent in a distributed system. The Distribution System Services (DSS) association, re-association and disassociation services determine the desired recipient when sending a message. This fact leaves independent manufacturers the open forum to design their respective transceivers to accomplish distribution however they see fit [CBC06]. Comparative analysis, covered more in Section 4.3, concludes derivations between independent manufacturers with respect to packet response. The deviations are attributable to the open implementation specifications set forth by IEEE.

2.4 Frame Classes in IEEE 802.11

There are two variables that all stations must manage, defined by the 802.11 standard. These variables, authentication and association, categorize the Station (STA) into one of three potential states. The standard further defines a series of class frames that each state may invoke. Depending on the associated state, these frames provide rudimentary service functions for authentication and association. As the STA transitions from an unauthenticated/unassociated to an authenticated/associated state, the frames the STA is able to transmit increases. Sections 2.4.1–2.4.3 describe the three states in detail, including the corresponding class frames associated to each state. Figure 2.1 illustrates the relationship between the three states.

2.4.1 State 1: Unauthenticated, Unassociated.

State 1 represents the initial state that all STA's begin. In this state, the STA may invoke a limited set of frames that allow the STA to identify an Access Point (AP), Extended Service Set (ESS) or WLAN to associate with. In addition, State 1 grants the STA the ability to perform the necessary functions to authenticate to any of the aforementioned. Unsuccessful authentication denies the STA the ability to transition to State 2, where association takes place. The permitted frames exchanged in State 1 are indicated as "Class

1" frames, respectively. These frames provides STAs the ability to complete authentication services and identify APs on the LAN.

2.4.2 State 2: Authenticated, Unassociated.

A STA transitions to State 2 upon successful authentication in State 1. In this state, the class of permitted frames to exchange expands to include association, disassociation and re–association capabilities. In addition, a STA in State 2 may receive a de–authentication notification. Subsequently, upon receipt of a de–authentication notification, the STA returns to State 1. Class 2 frames include all of Class 1 frames, as well as the association capabilities and de–authentication notifications. Upon successful completion of association, the STA transitions to State 3.

2.4.3 State 3: Authenticated, Associated.

State 3 defines a STA that contains true boolean values for both authentication and association. State 3 permits the transmission of all frame types. Within State 3, STAs transition back down to State 2 upon receipt of disassociation notifications and State 1 upon receipt of de–authentication notifications. If a STA receives a frame from another STA that is not authenticated, the receiving STA transmits a de–authentication notification to the sending STA. This forces the unauthenticated STA to return to State 1. Similarly, an unassociated STA transmitting Class 3 frames will receive a disassociation notification to force the return to State 2 [OHP05]. During this research, the STA's that manage each of the transceivers utilize data delivery service to send and receive data. The trials conducted within this experiment operate in State 3. Section 3.4.1 – 3.4.3 elaborate on the experimental methodology that references data delivery services.

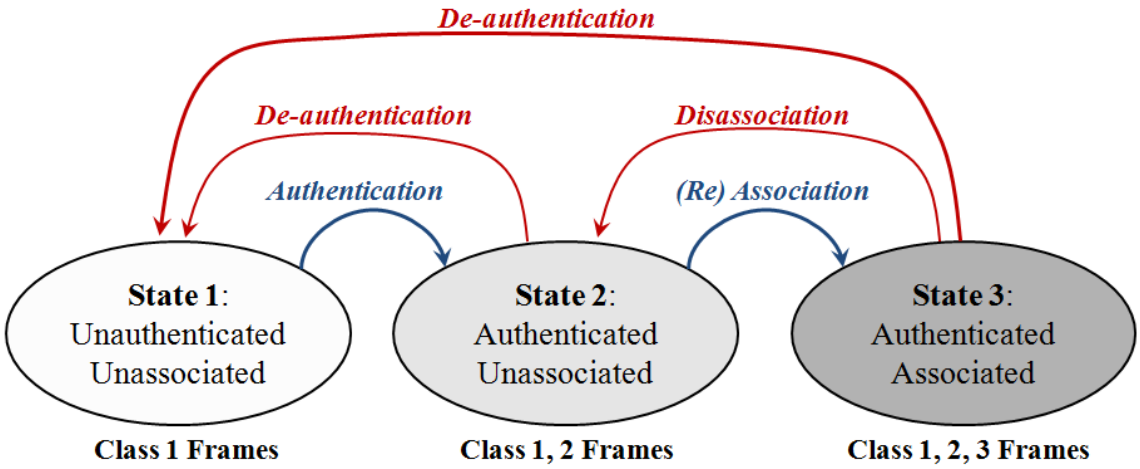

Figure 2.1: Relationship Between 802.11 States 1-3

2.5 Data Link Layer and Physical Layers

2.5.1 Sub-Layer Interaction.

There are multiple layers that pass data in the IEEE 802.11 standard. A unit of data takes on various names, depending on where the data resides on the computer networking model. Figure 2.2 depicts the relationship between the data link and physical layers. Data received from the network layer down to the Logic Link Control (LLC), the upper sub-region inside the data link layer, denotes the MAC Service Data Unit (MSDU). Before the MSDU is passed down to the Physical Layer (PHY) layer, a 30-byte MAC header and 4-byte Frame Check Sequence (FCS) encapsulate the MSDU, creating the MAC Protocol Data Unit (MPDU).

Once the MPDU traverses the MAC layer, a sub-region inside the data link layer, the MPDU enters the PHY layer, where it is referred to as the Physical Layer Service Data Unit (PSDU). The Physical Layer Convergence Procedure (PLCP) encapsulates the PSDU with the preamble and padding the same way the MAC header and FCS encapsulate the MSDU in the data link layer. This data unit, known as the Physical Layer Protocol Data Unit (PPDU), is managed and handled by the PHY layer [IEE99].

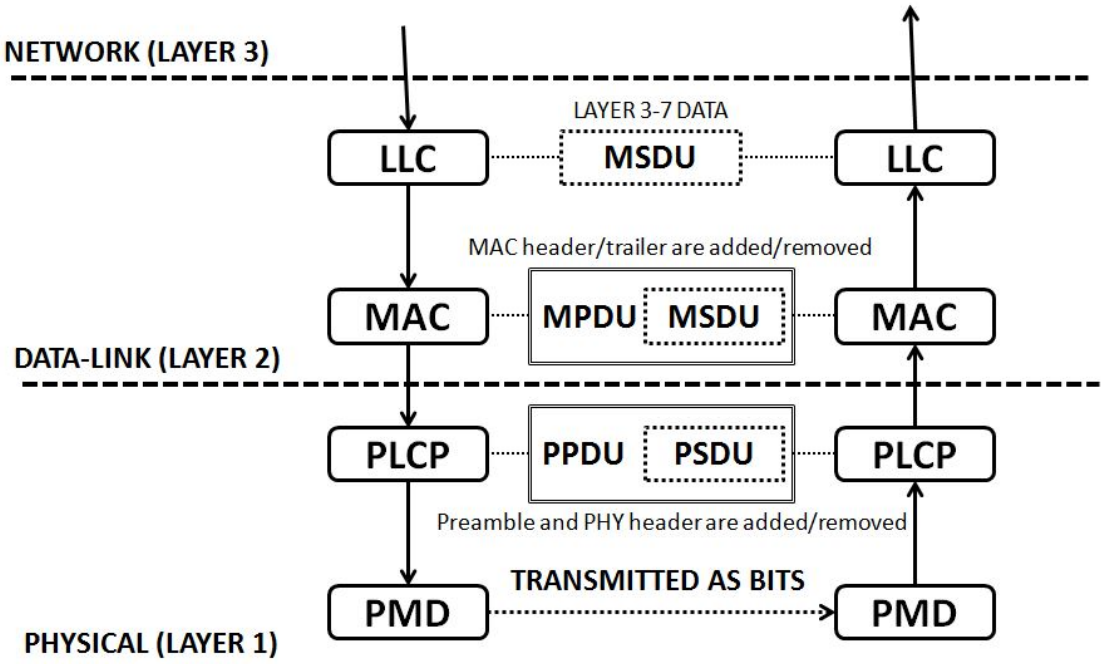

Figure 2.2: Relationship Between Network, Data-Link and Physical Layers

The PLCP and Physical Medium Dependent (PMD) sub-layers comprise the PHY layer. These two regions support PHY layer functionality similar to how the LLC and MAC sub-layers augment data link operations. The PLCP handles Clear Channel Assessment (CCA) and bridging MAC sub-layer communications. CCA techniques include Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) as well as Carrier Sense Multiple Access with Collision Detection (CSMA/CD). These functions regulate the nodes to ensure the availability of the channel before transmitting [FLM10]. The head of the PPDU contains the PLCP preamble, which sub-divides into the synchronization and Start Frame Delimiter (SFD) [Mol05].

2.5.2 Physical Layer Transmission Properties.

802.11b standard physical layer supports transmission speeds of 1 and 2 Mega–bits per second (Mbps). Utilizing DSSS, these speeds operate either through Dynamic Binary Phase Shift Keying (DBPSK) and Dynamic Quadrature Phase Shift Keying (DQPSK) modulation

techniques. DSSS also provides 5.5 and 11 Mbps speeds, by chipping the baseband signal with an 11-chip pseudo-noise code. As speeds near maximum throughput on 802.11b, the effective range of transmission diminishes as well as packet reception. Dynamic rate shifting techniques allow 802.11b to adjust the speed of transmission automatically, striking equilibrium between efficient throughput and minimizing packet loss.

2.5.3 Direct-Sequence Spread Spectrum PLCP Frame.

802.11b traffic follows the DSSS model. This transmission method differs from FHSS and OFDM. Figure 2.3 illustrates how DSSS transmits the RF signal over the range of a spectrum, rather than a narrow-band signal. The DSSS PLCP frame contains the preamble, header and PSDU. All of these fields process through a scrambling algorithm prior to transmission as the PMD, as opposed to FHSS models that only scrambles the PSDU (MAC frame).

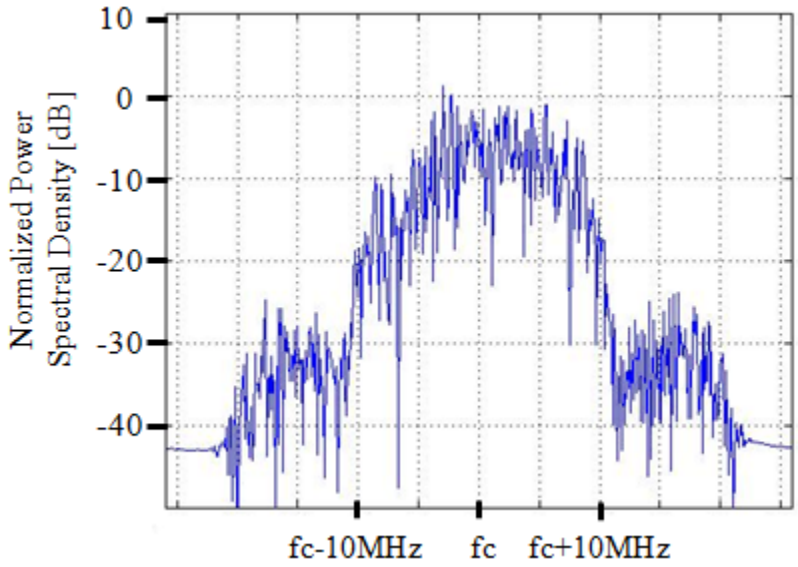

Figure 2.3: Example of DSSS Frequency Response [Ell08]

In 802.11b, DSSS transmits the preamble through either long or short synchronization. The former contains a 144-bit preamble, which comprises of a 128-bit synchronization field

followed by the 16-bit SFD. All 802.11b devices must support standard long preambles. The alternative shorter preamble reduces the synchronization field to 56 bits, which provides greater efficiency for specialized network traffic such as VoIP or real-time media streaming. Most 802.11b devices also support the shorter preamble standard [Erg02].

Within the standard long preamble, the synchronization field, SFD and PLCP headers transmit through DBPSK at 1 Mbps. The remaining traffic after the preamble and header transmits at rates best suited for their environments [MEG08]. Higher reliability, albeit lower throughput, include DBPSK (1 Mbps) and DQPSK (2 Mbps). Complementary Code Keying (CCK) or Packet Binary Convolution Coding (PBCC) enables the highest throughput of 5.5/11 Mbps.

Figure 2.4 depicts the packet structure of the long preamble this experiment utilizes. The long synchronization field of the preamble comprises of 128 scrambled bits. The SFD indicates the end of the preamble through a specialized sequence of bits, 1111 0011 1010 0000. To avoid ambiguity, the order of transmission reads from right to left as a result of least-significant bit. The long synchronization fields SFD differs from the short synchronization field by reversing the value of the long SFD [Gas05].

Following the preamble, the PLCP frame contains the four-field header that specifies information about the packet. The signal field is an 8-bit field that identifies the speed of transmission for the packet. Long preambles contain four possible values: 1 Mbps, 2 Mbps, 5.5 Mbps or 11 Mbps. The service field carries little relevance to the impact of this experiment, however this 8 bit field contains the values of clock lock and modulation techniques. Additionally, the last bit in the service field serves as an extension to the length field, which specifies how many microseconds are required to transmit the MAC frame. Lastly, the CRC creates the checksum as a result of the values within the PLCP header. This ensures the integrity of the header remains intact to signal lost bits during transmission [CSS11].

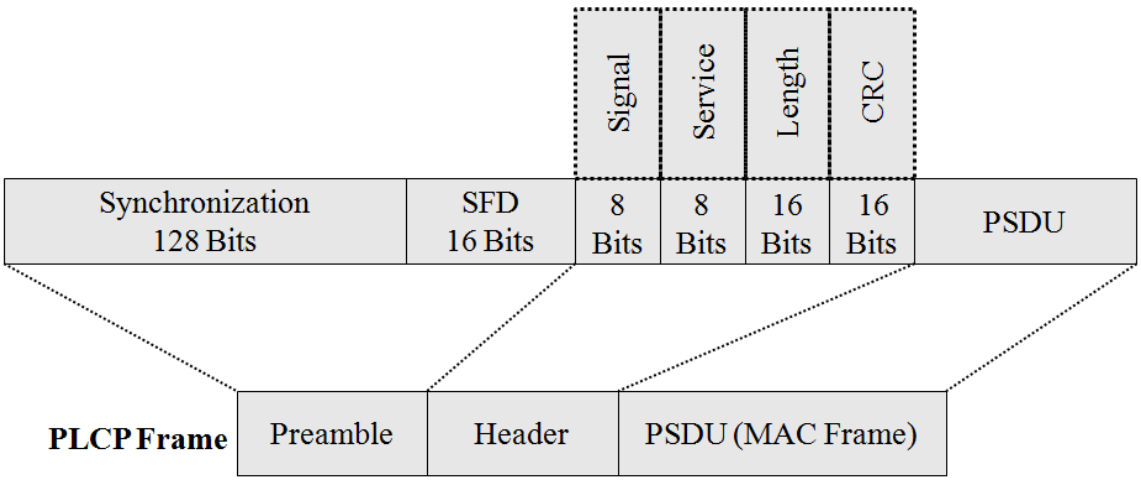

Figure 2.4: PLCP Long Preamble Structure [IEE99]

2.6 National Instruments Universal Software Radio Peripheral

National Instruments produces various software defined radios that provide record and playback capabilities. The USRP provides a cost effective technical solution to conduct experiments, while maintaining the integrity of the RF signal. As opposed to other hardware devices that offer comparable performance, the USRP's small size lends to portability and enables testing from the convenience of personal workspace. Housed inside the USRP–2921, the transceiver operates a direct conversion transmitter and receiver. The USRP–2921 model operates within the 2.4–2.5 GHz and 4.9–5.9 GHz range, the two RF spectrums utilized by IEEE 802.11 protocols [IEE99].

Section 3.3 discusses specific implementation, but the overall capabilities of the record and playback follow three primary components: $p(t) = r(t) = s(t)$. $p(t)$ represents the playback signal, which the host machine provides, running the LabView software that manages the USRP. $r(t)$ represents the recorded signal captured by the USRP device. $s(t)$ represents the desired signal to record. Ideal conditions preserve the integrity of

the recorded signal and the playback signal, however various sources produce external interference.

Accounting for interference that exists in the wireless medium, the previous equation transforms into: $p(t) = n_S(t) = s(t) + n(t) + i(t)$. $p(t)$ still represents the playback signal. However, $s(t)$ now contains noise present on the channel as well as environmental interference. Additionally, the actual devices themselves also produce noise, accounted for by $n_S(t)$.

Noise takes on multiple forms in the electromagnetic spectrum. Noise occurs from poor shielding around cables, present between the host machine managing the USRP and the actual device itself. This effect also occurs around loose connections and other antennas lacking proper configuration. Figure 2.5 represents how other devices operating within relative channel and physical proximity produce noise, known as co–channel interference, which degrades the quality of the signal. Bluetooth devices also operate within the 2.4 GHz - 2.4835 GHz range. Wireless devices operating strategically on channels: 1, 6 and 11 reduces the amount of co–channel interference. However 802.11b cannot avoid potential interference with Bluetooth devices, which employs FHSS.

In addition to noise on the channel, environmental interference also impacts the quality of the signal. Multipath propagation occurs when segments of an RF signal take paths of varying lengths prior to reaching the receiver, generally as a result of a physical obstruction or reflection off a surface. Multipath propagation also occurs in mobile networks where the sender or receiver are mobile. This phenomenon deteriorates the signal, producing a blurring effect once reaching the receiver. Lastly, although not relevant to this research, signal quality decreases when transmitted to distant receivers.

During recording, the USRP captures the playback signal with external interference. The USRP contains a fixed 40 MHz internal input bandwidth, driven by the 20 MHz antialiasing lowpass filters on the “I” and “Q” values. These two values represent the in–

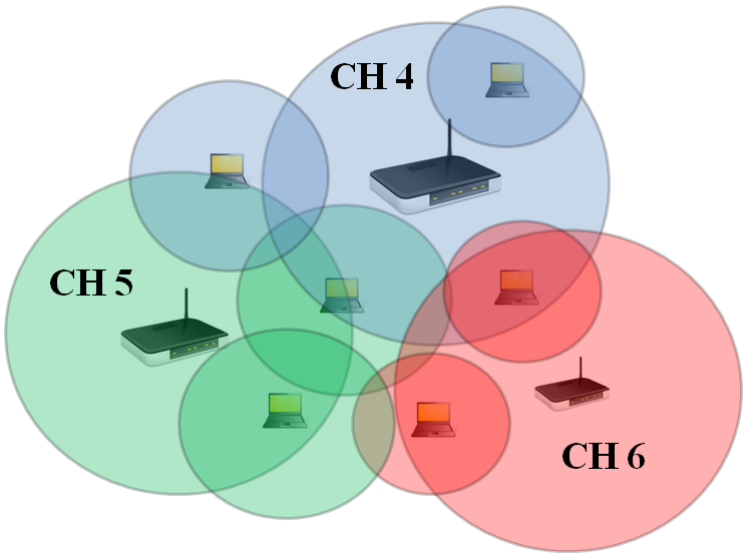

Figure 2.5: Diagram of Co–Channel Interference

phase and quadrature values during the digitization phase. During the digitization phase, the recorded signal is transmitted through the gigabit Ethernet cable where the host machine writes the data. LabView saves each sample, which includes the "I" and "Q" values, as 16–bit integers inside a vector. Sections 3.4.1 and 3.4.2 further cover the processing and plotting of data to perform preamble modifications.

In addition to external interference, the recorded RF signal experiences subtle deterioration as a result of physical limitations associated with the USRP. As indicated in Section 2.5.3, 802.11b operates through DSSS, which utilizes the entire band. DSSS comprises of the primary signal known as the "main lobe", and sideband lobes that transmit at reduced power to mitigate adjacent channel interference. Spectral density depends on the distance from the center frequency, which is defined by the IEEE standard. Transmissions within 11 MHz of the frequency center (main lobe) remain unfiltered. Sideband lobes contain the secondary signals adjacent to the main lobe, which operate between 11–22

MHz away from the frequency center. These filtered signals transmit at –30 Decibels (dB). Anything beyond ± 22 MHz from the frequency center transmits at –50 dB.

Figure 2.6 illustrates an RF sample utilizing DSSS to denote the region of potential loss of signal. The relationship between spectral density and the distance from the frequency center illustrate that DSSS transmits most of the data within the main lobe. Sideband lobes extend ± 22 MHz away from the center frequency, however the USRP is physically limited by the 20 MHz antialiasing lowpass filters. This means that there exists a small region at both ends of the signal where signal loss occurs, reducing the quality of the recorded and playback signal [Nat12].

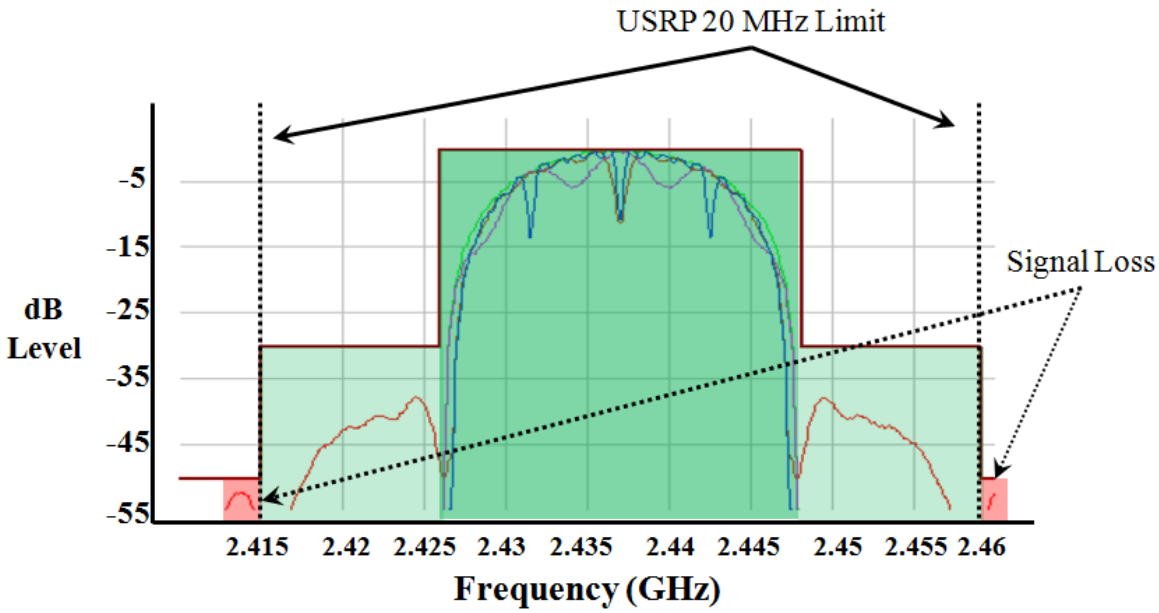

Figure 2.6: USRP Physical Limits on 802.11b Spectral Mask

2.7 Security Concerns Stemming from Arbitrary MAC Address Assignment

The wireless spectrum is a contested medium that offers portability and mobile network computing. Abuse of unique digital identifiers, such as MAC addresses,

introduces enticing conditions that lead to cybercrime. The resulting anonymity of masquerading legitimate devices lends itself to the assumption of avoiding legal repercussions. Cybercrime costs businesses millions in revenue and trends indicate an increase in abuse of cyber through Ethernet and wireless spectrums [Pon12]. Arbitrary MAC address assignment raises concerns because it affects the information security model: confidentiality, integrity and availability. Earlier detection of MAC spoofing provides system administrators an opportunity to isolate the threat and prevent a myriad of potential attacks, enabled through the abuse of unique digital identifiers.

2.7.1 ARP Cache Poisoning.

Man–in–the–Middle (MITM) situations can occur as a result of a successful Address Resolution Protocol (ARP) cache poisoning attack. There are other possibilities also directly attributable to MITM operations. Local Area Networks (LAN) employ ARP, which manages the list of Internet Protocol (IP) addresses to MAC addresses. Any client can initiate an ARP request to find where to appropriately route LAN traffic. Security concerns stem from the fact that the ARP protocol performs absolutely no authentication when a host receives an ARP reply. This creates a situation where any client connected to the LAN can issue arbitrary ARP replies that create a MITM attack. An attacker can subsequently sniff traffic passed between hosts any other host including the gateway [FlD01].

ARP cache poisoning also leads to potential denial attacks through MAC flooding. Figure 2.7 depicts both normal network conditions and the MITM attack. An attacker bombards the LAN with a series of ARP replies that cause a switch to act as a hub. This broadcasts all network traffic to all hosts associated with the LAN, which defeats port security implementations. Alternatively, the attacker can launch a denial of service attack by creating a half-duplex connection between the clients and gateway, disrupting services on the LAN [BXY+06].

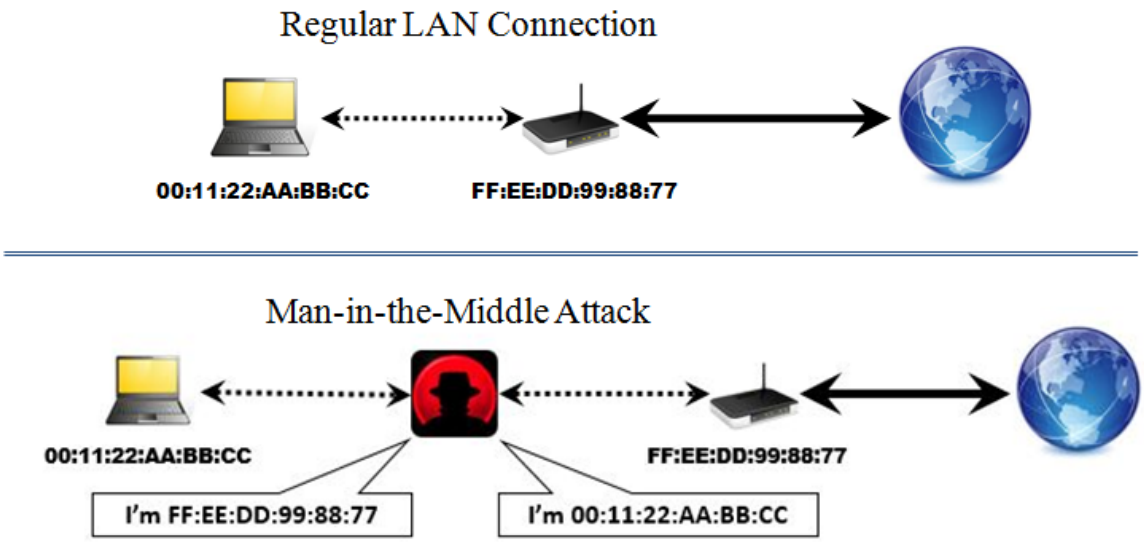

Figure 2.7: Diagram of Man-in-the-Middle Attacks

2.7.2 Domain Name System Poisoning.

Attackers that successfully masquerade as legitimate users through MAC spoofing techniques further gain leverage through Domain Name Service (DNS) poisoning. At a high level, the attacker spoofs their MAC address to gain access to a wireless network. A MITM attack allows the attacker to bridge communication between the victim and access point. DNS poisoning occurs when the victim enters a DNS request, but is given a malicious IP. An attacker employs the use of a proxy to intercept specific web traffic and change the properties before forwarding the response back to the victim. Burp proxy is a graphical user interface–based tool that provides such a capability [Bur14]. Additionally, the attacker may be able to win the race to respond to a DNS query and provide a malicious IP before the legitimate DNS response is received. In either event, DNS poisoning enables the attacker to pivot and further gain leverage onto a network [TVP10].

2.8 Related Research

There exist related research efforts that explore RF fingerprinting techniques for security purposes. Each method contains strengths and weaknesses to detect, isolate and mitigate potential attacks. Areas of research include Received Signal Strength Indicator (RSSI)–based analysis, Sequence Number Rate Analysis (SNRA) intra–cellular security using RF fingerprints, and device enumeration in IEEE 802.15.4 wireless protocols [RMT13]. In addition, spectral fingerprinting research efforts conducted in IEEE 802.11a yield moderately successful results analyzing SNRA [STM08]. Research efforts focus on the performance of physical hardware specifications in non–transient networks to contrast transceivers under test.

2.8.1 Intra–Cellular Security and RF Fingerprints.

Air monitoring applications provide additional security using RF fingerprints. Cellular wireless communications utilize the same contested medium, posing threats through unauthorized use of unique digital identities. The security paradigm hinges on the technical aspect of physical layer security, since spoofing is extraordinarily difficult. Related research indicates unintentional changes to the signal as a result of hardware implementation and component manufacturing. These changes include characteristics that affect frequency, phase and amplitude. Statistical analysis compares the characteristics of each of the manufacturers between standard deviation, variance, skewness and kurtosis. This related research concludes that intra–cellular security is feasible, citing similar methodologies employed in OFDM–based 802.11a signals analysis [RTM10].

2.8.2 MAC Spoofing Detection through RSSI.

The essence of RSSI research includes multiple sensors within a LAN that observe signal strength received from a transceiver, following packet transmission. Figure 2.8 models a simple RSSI diagram. Each sensor shares their data with adjacent neighbors to create an RF fingerprint based on observed signal strength. If a sensor receives an RF

fingerprint that is inconsistent with the characteristics of a known device's fingerprint, this triggers an alert. The device in question may be quarantined until further analyzed by a network or system administrator.

Variations exist on received signal strength, largely due to environmental interference. Factors that impact the quality of the signal include other communicating devices operating on the same channel, obstructions between two receivers, atmospheric conditions, even the curvature of the Earth. Irrespective the external interference, in a LAN where the sender remains in a stationary position, multiple samples from the RSSI produce a valid RF fingerprint. The RSSI model compares these fingerprints and performs real-time statistical analysis to discern differences in the behavioral properties to detect potential MAC spoofing. Based on computational algorithms, RSSI models create a "dB profile" to serve as a baseline for acceptance [STC08].

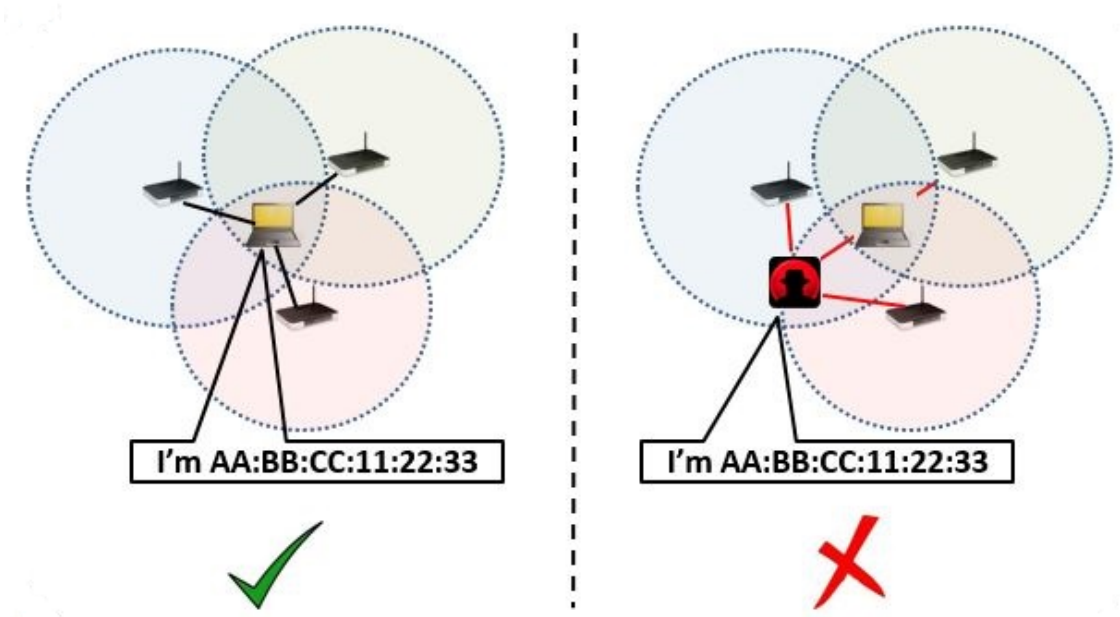

Figure 2.8: Example of Received Signal Strength Indicator Model

RSSI models demonstrate favorable properties for RF fingerprinting. The RSSIs prove difficult if not impossible to impersonate, since attackers do not have enough control over the signal strength to compensate for environmental conditions. Depending on the distribution of the sensors within the model, RSSI possesses strong correlation to the physical location of a transceiver [MGS10]. Network and system administrators control the sensitivity of the RSSI model, providing a fulcrum between security and functionality. Lastly, the model performs with high consistency on stationary transceivers. This significantly reduces variation on dB levels of RSSIs and mitigates false positive readings.

Limitations exist within the RSSI model, especially with mobile computing. Transceivers that operate on mobile devices prohibit the ability for sensors to construct valid RF fingerprints. This differs from stationary platforms, since sensor networks obtain the same RSSI values when conducting multiple tests. As such, RSSI–based approaches work best in static environments where the locality of each station remains consistent [Wri03].

2.8.3 MAC Spoofing Detection through SNRA.

SNRA–based methods analyze a key field within an IEEE 802.11 frame. The sequence control frame contains a 16-bit field divided between the 12-bit sequence number and the 4-bit fragment number. The IEEE 802.11 frame employs a standard FCS that detects corrupted or altered bits. Additionally, the ability to generate arbitrary sequence numbers in IEEE 802.11 traffic is not as trivialized as changing the MAC address on a wireless transceiver.

The 12-bit sequence number field equates to a range of values that initialize at zero and increment until 4095, before wrapping around to zero. Successfully received packets increment the sequence number, which facilitates the filtering of duplicate traffic on the wireless network. This predictable pattern of incrementing sequence numbers yields

advantages to both network attackers and defenders. SNRA–based methods analyze the differences in sequence numbers to detect MAC spoofing. A simple algorithm calculates the differences between two frames and accounts for the wrap-around that occurs once the sequence resets from 4095 to 0. Lastly, the detection algorithm provides a measure of flexibility in the event that packets are received out of order.

For example, if the detection algorithm sets the out-of-order threshold to 4090 or greater. Any differences in sequence numbers between two frames greater than 4090 are treated as legitimate traffic that arrived out of order. Naturally, there exists the possibility that MAC spoofing occurred relative to these values. However, based on 10,337 registered prefixes out of a total possible 16,777,216 MAC prefixes, the odds of MAC spoofing activity based on random sequence number assignment stands at 0.0006% [IEE14].

Two known methods exist to extract the sequence number from the MAC header. The first method includes the modification of the WLAN interface driver for both the access point and all associated stations. This method allows for both the detection and prevention of MAC spoofing, however the primary drawback is the modification of firmware in a commercial access point requires extensive technical knowledge. Employing a WLAN monitor offers an easier method to leverage the sequence number from the MAC header. When these systems operate in monitor mode, they receive all IEEE 802.11 traffic, irrespective the operating channel. Consequently, these monitor system that operate separate from the network do not possess the capabilities to prevent MAC spoofing [GuC06].

MAC spoofing detection through SNRA–based techniques monitor sequence number gap differences similarly the way this research isolates and identifies 802.11b bursts of traffic in MATLAB. A difference that does not meet the criteria of a wrap-around scenario or out-of-order sequence strongly suggests that MAC spoofing has occurred. Ultimately, the ability to avoid detection from SNRA based techniques requires extensive knowledge

of the target's sequence number and precision timing techniques to align with anticipated IEEE 802.11 traffic.

The primary limitations of SNRA–based techniques include the overhead cost of maintaining software to track assigned MAC addresses to their current transmitted sequence numbers. Similar to RSSI models, mobile networking introduces challenges with SNRA approaches. Clients that roam out of the range of the network discontinue transmitting their respective sequence numbers. When the clients return, SNRA–based models observe a jump in sequence numbers, which triggers a response of potential malicious traffic. Timing delays mitigate false positives by establishing a threshold from which a client re-associating with the network is assumed to have roamed out of range for a temporary period of time [GuC06].

2.8.4 MAC Spoofing Detection through Secondary Authentication.

Secondary authentication methods provide alternative security approaches to address the issue of arbitrary MAC assignment, a flaw in the data–link layer of the OSI model. Related research explores security models that augment intrusion detection by comparing the hashes of legitimate users against unknown users. In this process, a legitimate user creates a psuedo–fingerprint that identifies properties unique to the user such as: MAC address, central processing unit identification and computer name. The user combines these properties, generates a hash and passes the authentication frame to the access point.

When a new user requests to associate to the network, they pass an authentication frame to the access point. Network association occurs if the access control list validates the hash of the authenticating frame. Additional security procedures include periodic re–authentication to the network and disassociation from the network after multiple failed re–authentication attempts. While the security model elevates the level of difficulty to spoof legitimate MAC addresses, re-authentication lockouts create denial of service opportunities against legitimate users on the network. [BhA12]

2.8.5 MAC Spoofing Detection through Prefix Validation.

IEEE maintains the vendor prefix that comprises the first three octets in a MAC address. This list is similar to DNS lookups, which provides registration information [IEE14]. Simple scripting languages obtain the MAC address of clients that connect to an access point from a predefined dictionary updated by IEEE. MAC validation prohibits the use of randomly assigned MAC addresses from authenticating to the network. This methodology provides very minimal security for networks, since MAC validation only asserts that the first three octets of a client exist on the organizationally unique identifier maintained by IEEE.

III. Methodology

3.1 Introduction

This chapter covers the experimental setup and design configurations to enumerate the RF fingerprint of each device. Section 3.2 describes the problem definition, goals, approaches, system boundaries and system services. Section 3.3 explains the experimental design, workload parameters, system parameters and factors. Section 3.4 describes the design and configuration of the experiment. Section 3.5 summarizes the overall methodology.

3.2 Problem Definition

3.2.1 Goals.

This experiment explores the possibility to perform RF fingerprinting by analyzing transceiver response rates to a series of modified preamble trials. Initial hypothesis suggests that shortening the preamble of 802.11b traffic reduces the transceiver's ability to perform network synchronization. This causes the transceivers to respond differently. Fingerprinting each difference in response can be used to identify each device. The overarching goal is to correctly identify a unique transceiver communicating with a wireless AP operating in 802.11b. Successful identification is contingent upon several questions that this experiment addresses:

1. Does there exist the ability to capture, alter and replay 802.11b wireless traffic?
2. Do wireless transceivers respond differently to altered wireless traffic?
3. If so, are the responses unique enough to perform device classification?

3.2.2 Approach.

The system, named the Signals eXploitation System (SXS), addresses each goal by creating a small wireless network and performing controlled tests on each of the

transceivers. Several integral components provide specific functions throughout the trials. An arbitrary waveform generator (National Instruments USRP–2921) performs the packet capture and replay. MATLAB software displays a graphical user interface to interact with the captured data. Further, MATLAB is used in this research to alter the vector representing the RF fingerprint. Combining these elements, SXS tests how the transceivers respond to varying preamble lengths. The results of packet response to each of the trials determines whether or not device classification is possible.

3.2.3 System Boundaries.

Several components comprise the system under test of SXS. The SXS manages the system parameters of the wireless settings used during experimentation. SXS also manages each trial containing the modified preambles as part of the workload parameters. Components within SXS include: the Ettus USRP–2921 and corresponding laptop that manages the LabView software, the transceiver under test, and the wireless access point. Other components in SXS include the wireless protocol, channel, access point configuration, USRP configuration and wireless protocol analyzer.

This experiment restricts the scope of this research analysis to 802.11b, which employs DSSS modulation. Techniques applicable to 802.11b differ from other protocols, such as IEEE 802.11a, that may employ OFDM or FSSS instead of DSSS. Additionally, 802.11b operates exclusively within the 2.4 GHz Industrial, Scientific and Medical (ISM) band, whereas other protocols may transmit in the 2.4 GHz or 5 GHz range. Lastly, SXS operates exclusively on channel 6 throughout the experiment.

3.2.4 System Services.

The SXS provides transceiver enumeration capabilities. SXS alters the number of bits in the 802.11b preamble by injecting noise and replacing the RF signals that represent bits in the preamble. Afterwards, SXS retransmits these RF signals that contain the modified preambles to gauge packet response in each of the tested transceivers. Each

experiment tests the transceivers ability to respond to packets with truncated preambles over the course of 10 trials. The number of trials adequately illustrates how transceivers fail at various stages during the experiment. Two transceivers fail with four or less bits removed, four transceivers fail between five and nine bits removed and two transceivers continued to transmit even with 10 bits removed. This approach exploits independent hardware manufacturing in packet response to ICMP packets. Differences in packet response suggests device classification is possible. By verifying the MAC addresses with their observed performance during the experiment, this methodology creates an additional layer of security that thwart the abuse of digital unique identifiers.

3.3 Experimental Design

Figure 3.1 diagrams the overall setup of SXS, including the individual components. Section 3.3.1 addresses the workload parameters. Section 3.3.2 details the performance metrics. Section 3.3.3 covers in depth the system parameters. Section 3.3.4 discusses relevant factors applicable to this research.

3.3.1 Workload Parameters.

The primary workload parameters of SXS are the bits removed from the preamble. This parameter affects all 802.11b wireless traffic since the preamble is prefixed during the PLCP. The preamble allows the sender and receiver to perform network synchronization, channel estimation and frequency offset functions. Figure 3.2 illustrates how the RF signal appears after the preamble modification by injecting noise. The top data plot illustrates a trial with one bit removed from the preamble, whereas the bottom data plot illustrates a trial with two bits removed from the preamble. The noise region in the bottom data plot is distinctly longer, since more bits are removed from the preamble. Removing initial bits from the preamble decreases the two communicating transceivers window to complete synchronization. Depending on hardware specifications, synchronization fails if too many

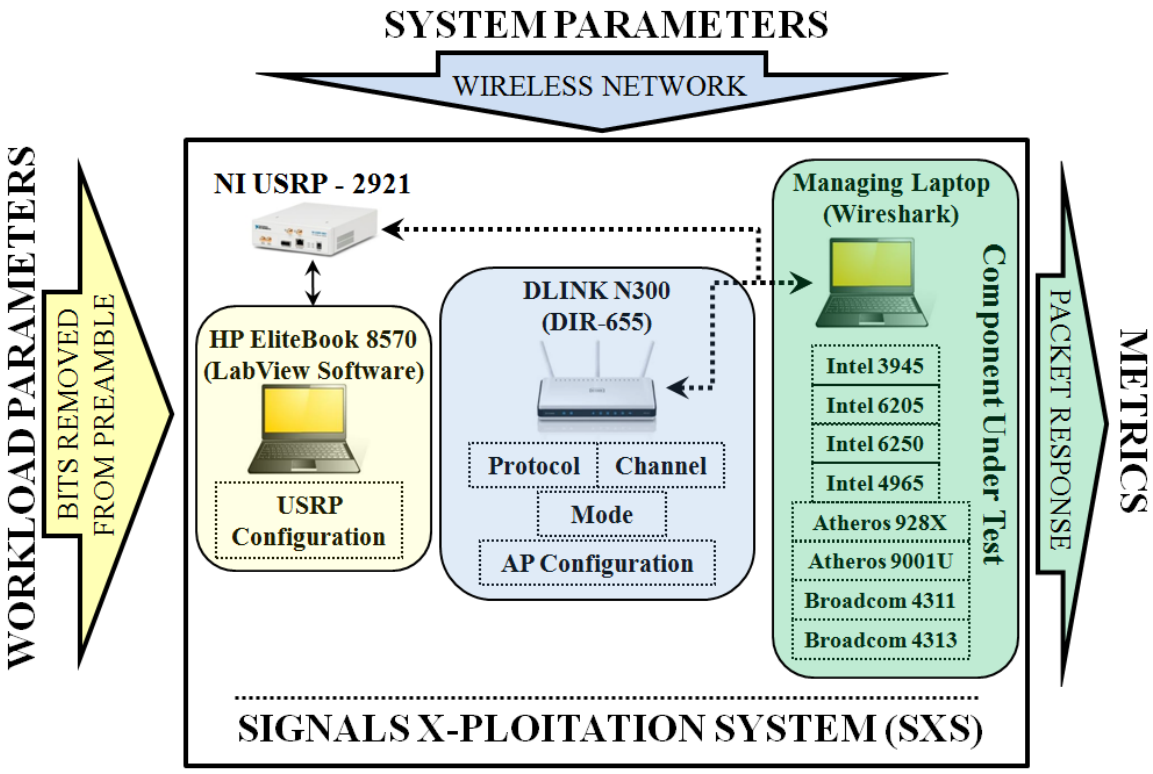

Figure 3.1: System and Component Under Test Diagram

bits are removed. Section 3.3.2 addresses how performance metrics are derived from the workload parameters within SXS.

3.3.2 Performance Metrics.

The overall response rate of the transceivers to the series of trials represents the performance metrics used. SXS executes 40 total transmissions, which equals a single trial. Each transmission contains a test packet with the modified preamble. The transmission lasts for a duration of three seconds, buffered by an additional second before the next transmission. This ensures that each transmission does not interfere with one another. Conducting each trial with 40 transmissions ensures there exist adequate data to measure the performance of the transceivers through statistical analysis. Device classification compares the packet response rate of each of the transceivers throughout the experiment.

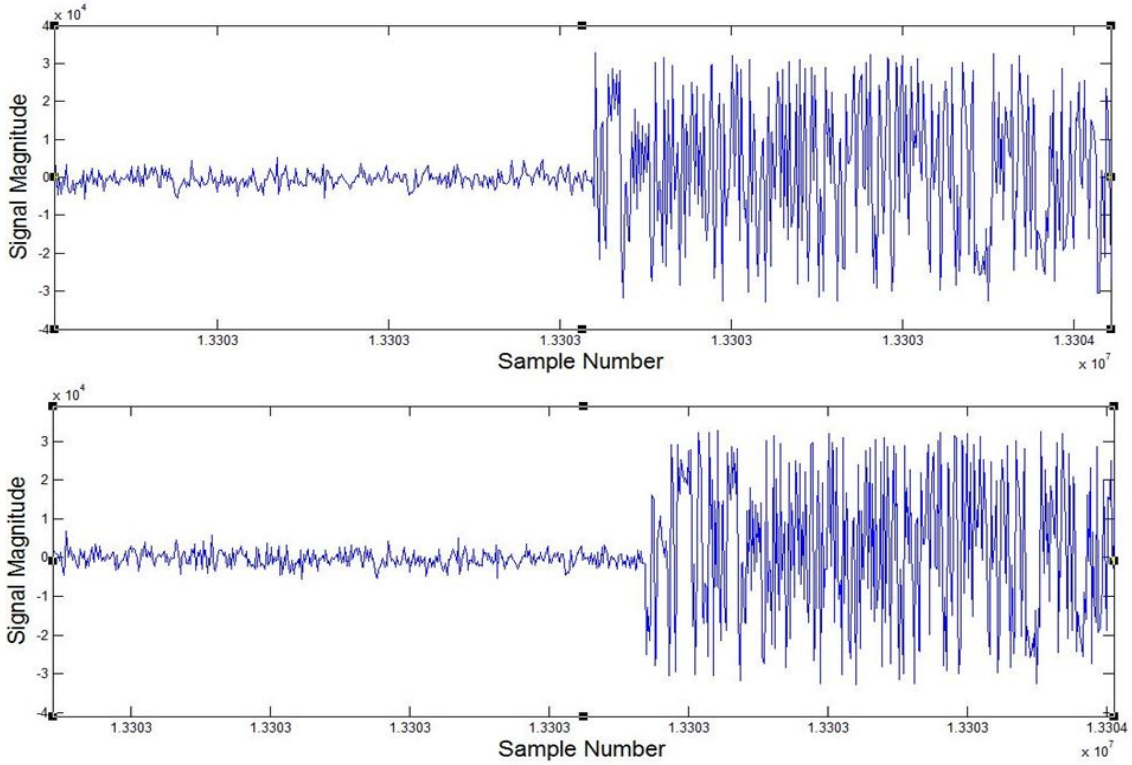

Figure 3.2: Contrasting RF Signals Between One and Two Bits Removed From Preamble

In some cases, a transceiver under test fails to respond to all 40 test packets, which is referred to as the"zero–barrier" threshold. This performance metric brings to light the fact that independent manufacturing, while adhering to the standards defined by IEEE, leads to devices that reach zero–barrier at different trials during the experiment.

Modifying the preamble provides multiple avenues to analyze transceiver performance, beyond the observation of zero–barrier. As more bits are removed from the preamble prior to complete failure, packet response decreases at varying rates. This provides an alternative angle to profile the performance of a transceiver, especially in cases where zero–barrier occurs during the same trial. Section 4.3.5 discusses this circumstance further in the case of the Atheros transceiver. Each iteration within a trial results in a binary response of either success (ACK) or failure (no response). This is a desirable characteristic for performance

metrics as it enables a series of statistical computations to validate the data. Most successful ACKs occur within half a second. However, transceivers have up to three seconds to respond before failure. Section 4.4.1 further explores the results of packet response and addresses the implications and assumptions derived from the experiments.

3.3.3 System Parameters.

The wireless network comprises the system parameters for this experiment. These parameters affect the performance of SXS. The characteristics of the wireless network include all supplemental hardware necessary to support the experiment. These include the routers, cables, network interface cards and actual sender/receiver.

The primary system parameter of SXS is 802.11b. This protocol provides a low–risk environment to implement the experimental design and adjust parameters as necessary to test the hypothesis. All commercial hardware that are "Wi-Fi" compliant support the 802.11b protocol. The resources to manufacture hardware compatible with 802.11b are cheaper than other wireless protocols, making it widely used in residential settings. Within 802.11b, the other system parameter that affects SXS include the channel of operation. Channels indirectly impact the quality of the RF signal. If several users utilize a particular frequency to transmit information, there exists a greater possibility of collisions. The modulation mechanism is another property of the wireless network. Although 802.11b only supports DSSS, other potential protocols may employ modulation schemes such as OFDM or FSSS. In addition, the trials are conducted with standard long preambles. Lastly, the wireless access point represents a system parameter as it an integral part of the wireless network. However, for purposes of SXS, the access point remains constant since the component under test is the transceiver.

The primary router passing traffic is the DLink N300 (DIR-655). The access point is white-listed to only accept association requests from devices with a specific MAC address. Further, the access point transmits on the 2.437 GHz band, operating on channel 6. The

network line speed remains at 2 Mbps, however it is worth noting the preamble actually transmits at 1 Mbps. The access points IP address is set to 192.168.0.1 and also assigns the IP address of 192.168.0.197 to each transceiver during the experiment. Only one transceiver is associated to the access point at any given time throughout the experiment.

Figure 3.3 depicts the 14 channels 802.11b contains. Each channel is 22 MHz wide and spectrally overlaps with adjacent neighbors. Popular channels include 1, 6 and 11 since none of these channels overlap with one another. The use of Channel 14 is prohibited in the U.S. and Channels 12 and 13 contain certain restrictions, not relevant to this research. SXS operates on channel 6 throughout this research.

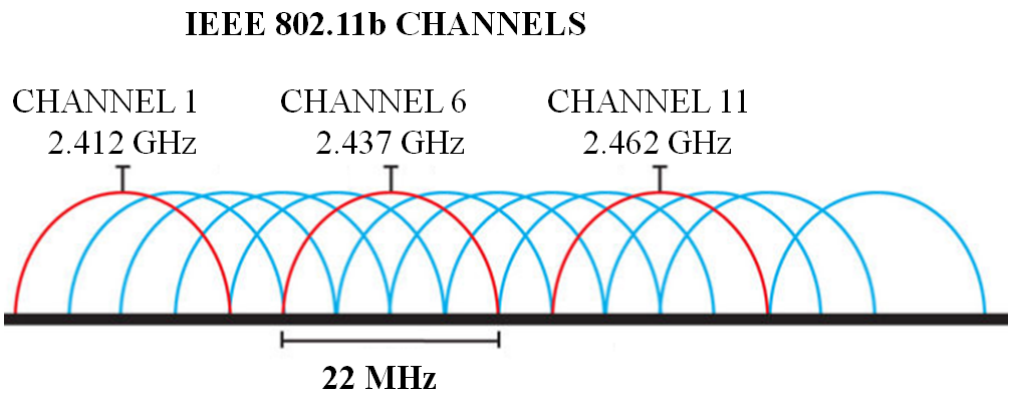

Figure 3.3: Three Primary Channels Utilized in 802.11b [IEE99]

Each computer uses a wireless network interface card during the experiment. The transceivers includes the Intel, Atheros and Broadcom wireless adapters. Some transceivers are built into the computer controlling the component under test as internal wireless adapters. Others, such as the Atheros 9001U-2NX (AirPcap) are USB adapters. These transceivers are capable of operating in promiscuous or monitor mode. This facilitates the capture of traffic, which is sent in three packet bursts using ICMP traffic. Only the first ICMP packet contains the modified preamble, while the other two ICMP packets remain

unmodified for control purposes. This allows for negative response and mitigates erroneous transceiver profiling, due to legitimate packet loss in a contested medium.

Since data collection mostly occurs in a UNIX environment, it is preferable that the tested transceivers have Linux driver support. More specifically, the Backtrack 5R3 suite contains tools that monitor wireless traffic through Wireshark and connectivity to APs through the wireless interface connection daemon. The only exception to this case rests with the Atheros 9001U-2NX (AirPcap) wireless adapter. AirPcap operates in a Windows environment and utilizes monitor mode.

3.3.4 Factors.

There are several factors in this research that have significant impact on the results of the experiment. Table 3.1 lists the direct experimental factors with their corresponding values. SXS contains three primary factors: preamble modifications, transceiver mode and transceiver model.

The preamble modifications range from removing a single bit up to 10 bits. Each successive trial removes an additional bit to gauge packet response from the tested transceiver. Preamble modifications prove ideal over other sections of wireless traffic. The preamble is encapsulated at the front during the PLCP. This characteristic eliminates the need to traverse legitimate wireless traffic and approximate where to overwrite the RF signals, covered in depth in Section 3.4.2. As the number of bits removed increases, the hypothesis states that the transceiver responses rate will decrease.

Transceivers are the key factor within SXS, represented as the component under test. Independent manufacturing yields behavioral inconsistencies while fulfilling services required by 802.11b specifications. Tools such as "Macchanger" provide arbitrary MAC address assignment in UNIX environments [Gnu04]. However, no current methodologies demonstrate the ability to alter the physical characteristics of a transceiver to mimic another. In other words, current open source software allows anyone to change a MAC address

with a single command. However, spoofing physical properties requires hardware and firmware modifications, lending itself impractical. This is critical, since the methodology depends on the integrity of the transceiver under test and assumes no physical or firmware modifications.

The only factor that encompasses a physical capability is the mode of operation, observed by the two Atheros transceivers. Neither mode of operation (promiscuous, monitor) introduces any significant anomalies. Similarly, while the majority of transceivers operate within a laptop, the Atheros AR9001U-2NX is a Universal Serial Bus (USB) wireless adapter.

Table 3.1: List of Experimental Factors

Factor	**Level**
Preamble Modifications	1–10 Bits Removed
Transceiver Mode	Promiscuous, Monitor
Transceiver Model	Atheros–Based Broadcom–Based Intel–Based

3.4 Experimental Methodology

Two of the most important factors include the transceivers model and preamble modifications. Eight different transceiver models, manufactured by three different companies allow for various comparative analysis. The packet response rates during the experiment illustrate the characteristics of each transceiver. The trials consist of a control ICMP echo request and two unmodified ICMP echo requests. In other words, each transmission within a trial contains three total pings used to discover a live host on the network. The first ping contains the preamble modification to account for a positive

response. Two unmodified pings accompany the control ping to account for negative responses. Table 3.2 lists the three possible outcomes when executing a single transmission:

Table 3.2: List of Possible Experimental Outcomes

Positive Response:	Successful ACK to the modified ping.
Negative Response:	Successful ACK to unmodified ping, but not to the modified ping.
Network Error:	No ACK to either modified or unmodified ping.

Conducting this research in a live environment provides benefits and demonstrates practicality if SXS is deployed onto an actual network. If the experiment is conducted in an environment free of noise or other external interference, it increases the prospect of differing packet response rates of the transceivers compared to their performance in a live environment. This results in an increase in false positive events, decreasing the practicality of SXS.

The experiment modifies preambles through a three-step process. First, the USRP captures and intercepts the initial 802.11b traffic. Next, MATLAB analyzes and processes the traffic as a binary file. Last, the USRP retransmits the modified packets to the transceivers under test [KRM14]. Through replication, this process creates 10 sets of test packets, each set increases the number of bits removed from the preamble.

3.4.1 USRP Packet Capture.

A machine associates and authenticates with the DLink N300 (DIR-655) wireless router, as indicated in (1) of Figure 3.4. Wireshark, operating on the machine of the transceiver under test, observes the ICMP packets transmitted to the access point. Successful association and authentication enables data delivery services and the USRP records and captures a series of ICMP echo requests (2), Figure 3.4. The RF signals captured during the recording process pass through the 1 Gigabit (Gbit) Ethernet cable

connected from the USRP to the managing host machine, an HP EliteBook 8570 laptop (3), Figure 3.4. The binary file contains the instantaneous amplitude signatures of the RF signals generated by the transmitted packets (4), Figure 3.4. The total transmission time to record the RF signals is under one second, producing a 48 MB binary file.

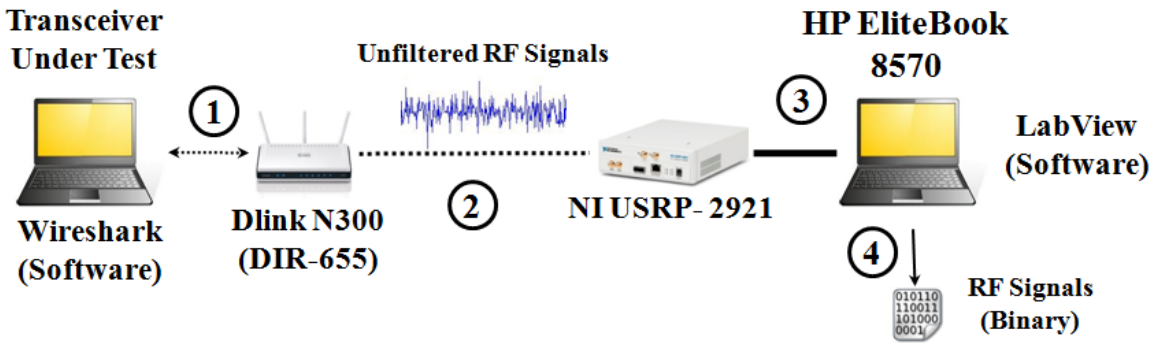

Figure 3.4: Experimental Setup to Capture RF Signals

An HP EliteBook 8570 laptop contains the LabView software that manages the USRP. LabView provides the graphical user interface to specify record and playback capabilities for the USRP device. In addition, LabView offers visualization capabilities to view wireless traffic and observe the quality of the signal. Labview assigns the USRP an IP address of 192.168.10.2. Since the tests occur on Channel 6, the carrier frequency is set to 2.437 GHz. 802.11b channels are 22 MHz wide. To ensure all 22 MHz of bandwidth transmits through the USRP, the IQ rate is set to 25 Million samples per second (Msps). Each sample contains an "I" and "Q" value, both of which are 16 bits in length. Therefore each sample is 32 bits. Transmitting 25 Msps results in $25,000,000(samples/second) * 32(bits/sample) = 800Mbps$. This requires a gigabit Ethernet port to successfully transmit the binary file to the USRP. 25 Msps provides the best resolution on 802.11b. As a result of hardware limitations between down conversions within the USRP and sustaining a 100 Mbps write

rate to the hard drive, 25 Msps is the maximum throughput achievable for this research [Nat12].

3.4.2 Preamble Modification.

The unaltered 802.11b traffic contains a multitude of RF signals not significant to this research. As shown in Figure 3.5, a Dell Precision T7500 desktop running MATLAB software obtains the file via a USB thumb drive to perform traffic analysis and isolate the transmitted packets. Preliminary traffic analysis examines the length of the RF signals to identify the original ICMP packet. The captured ICMP echo requests range between 26,000 – 26,024 samples in length. The correct sequence number associated to the three ICMP transmissions are 110, 111 and 112. Millions of samples comprise the vector. In order to determine the start of an RF transmission, MATLAB runs a for–loop script that traverses the binary file and returns the value of the sample that exceeds a predefined threshold. The aggregate process of refining the raw RF signals to modified preambles occurs during steps (6) and (7) in Figure 3.5. The next section further illustrates this process. Lastly, the HP EliteBook 8570 receives the binary file via a USB thumb drive to prepare for retransmission (8), Figure 3.5.

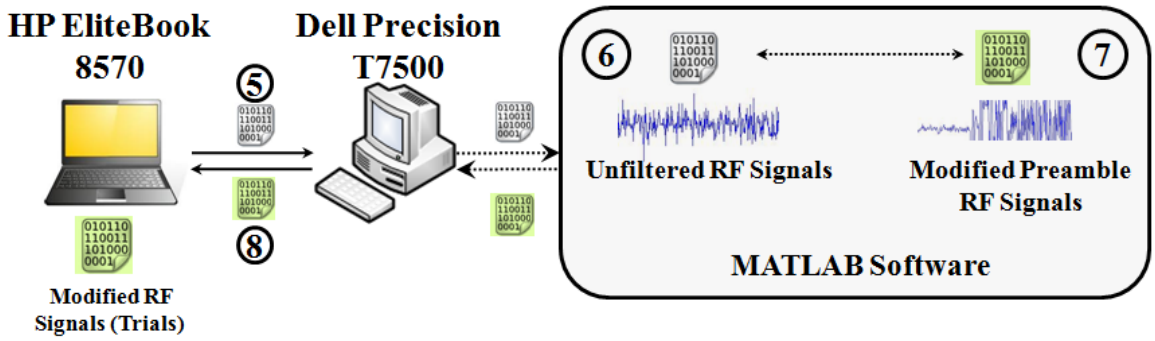

Figure 3.5: Modifying the RF Signals

Observing the instantaneous amplitude value using the Y–axis, MATLAB runs the for–loop script illustrated in Figure 3.6 in a restricted region where RF traffic occurs to find the index whose value exceeds 10,000 $\left(1 * 10^4\right)$. The restricted region is identified by observing the x–axis of the data plots, as illustrated in Figure 3.7. Instantaneous amplitude values in the noise region range anywhere from ±9,000. RF traffic amplitude values range anywhere from ±30,000. The for–loop script identifies when the amplitude values jump significantly outside of the values observed within the noise region.

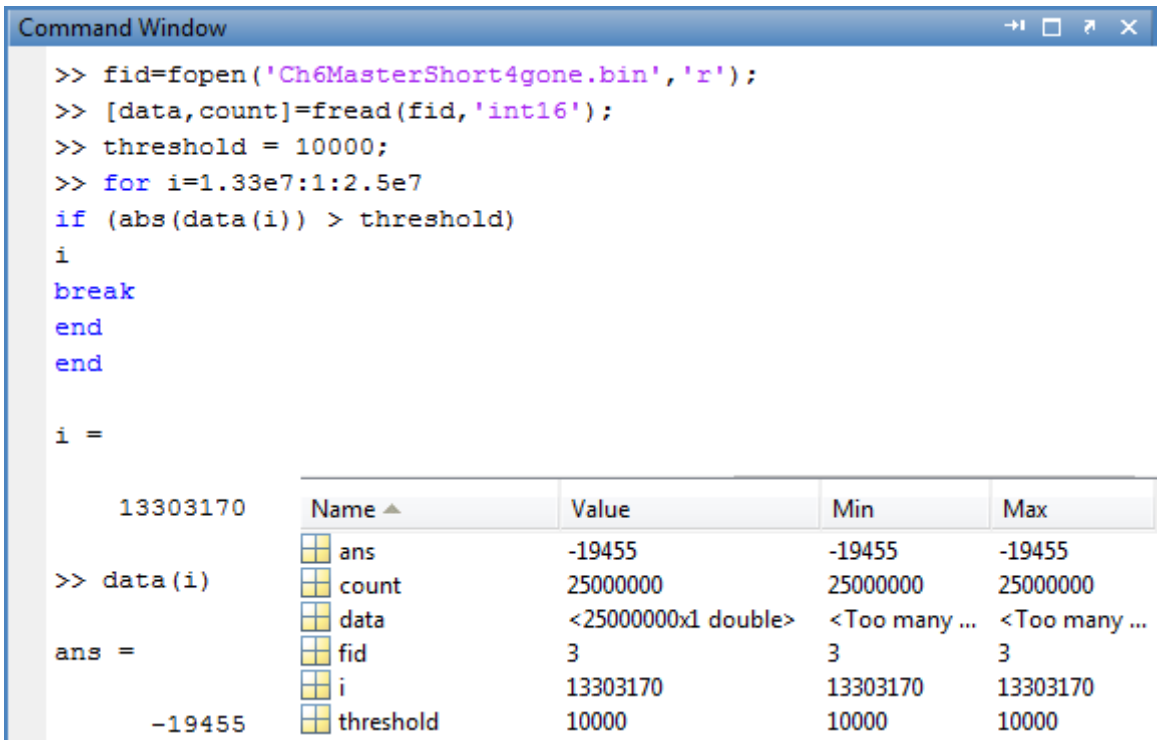

Figure 3.6: MATLAB Command and Workspace Windows Identifying RF Burst

Figure 3.7 illustrates the detection of an RF signal by contrasting the noise region with the burst region. After identifying and isolating the unknown RF signals, MATLAB creates a new binary file containing the sole RF burst. Wireshark monitors the interface on the machine of the transceiver under test. When the USRP replays the new binary file

created by MATLAB, Wireshark displays information about the unknown RF signal. If the unknown RF signal is the ICMP echo request recorded by the USRP, Wireshark verifies this by displaying a sequence number of either 110, 111, or 112. SXS repeats this process until identifying all three ICMP packets that contain either 110, 111 or 112 sequence numbers. This process is essential in determining the locality of the desired signals. MATLAB only retains the RF signals of the transmitted ICMP echo requests to construct the test packets for each trial.

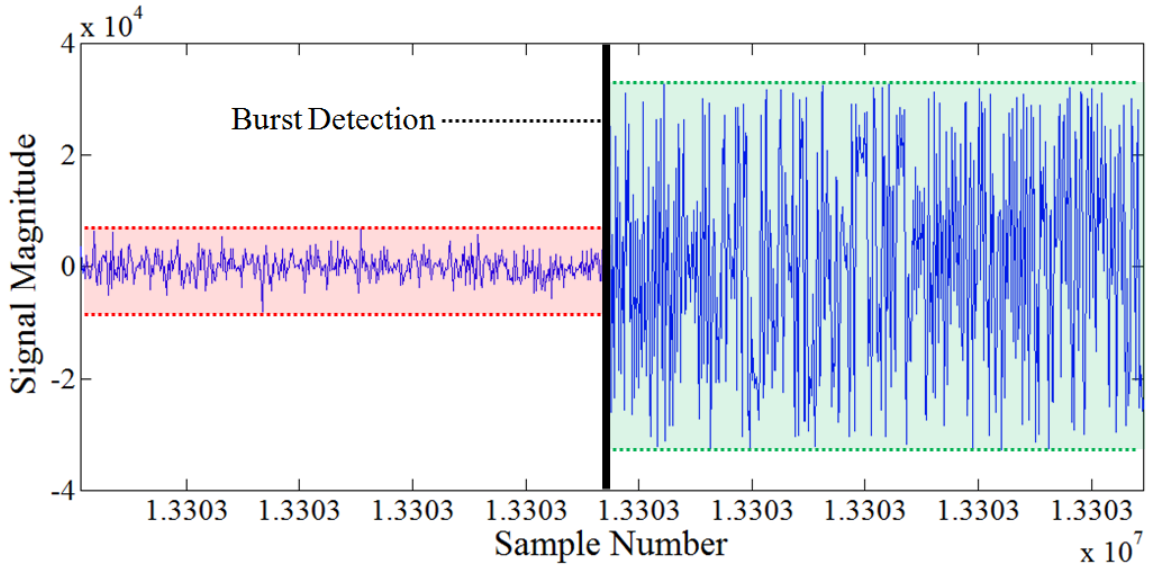

Figure 3.7: MATLAB RF Burst Detection Plot

Next, MATLAB concatenates the three binary files into a single transmission. Each transmission contains the three ICMP echo requests RF signals. The RF signals contain padding before and after, as well as noise that is injected between each transmission. The padding serves to provide between 100 – 150 milliseconds between the transmissions of each ping, similar to a guard interval. This mitigates propagation delay, echoes and reflections – side effects that 802.11b is sensitive to. Packet response is measured using

Wireshark on the machine of the transceiver under test. Only the first of the three ICMP packets contains the modified preamble.

3.4.3 Transceiver Configuration.

Each transceiver, except the AirPcap transceiver, connect to a standard commercial D-link N300 DIR-655 wireless router. The AP assigns the same IP address to each of the transceivers that connect, using 192.168.0.197. Further, the router only accepts association requests from transceivers by white–listing the MAC address. A series of ping requests verifies connectivity between the AP and the tested transceivers (#9), Figure 3.8. Recall in Section 2.4.3, data delivery service occurs once a machine successfully authenticates and associates to the network (State 3). The HP EliteBook 8570, containing each of the trials, passes the binary file onto the USRP–2921 through the 1 Gbit Ethernet cable. Wireshark, residing on the machine of the transceiver under test, monitors successful responses to the modified packets that SXS transmits through the USRP (#10), Figure 3.8.

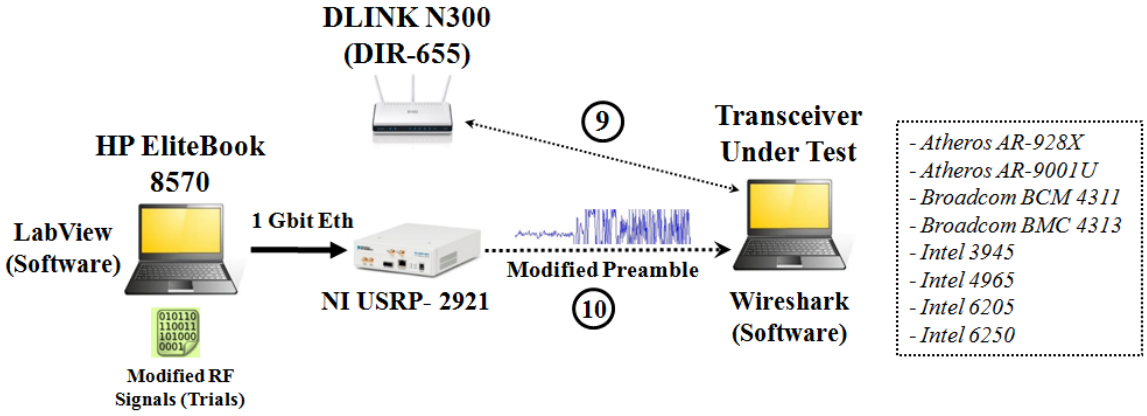

Figure 3.8: Transmitting the Modified RF Signals

Figure 3.9 displays the graphical user interface for the USRP settings utilized throughout the experiment. Packet replays occur at 2.437 GHz frequency, or channel 6 within 802.11b. Each set of packets resides on the HP EliteBook 8570, which passes

the binary file through the Gbit Ethernet cable. The "TX Destination File" field specifies the location of the test packets. The USRP and the receiving device rest between 6 – 12 inches apart, which mitigates packet loss during transmission. Furthermore, the USRP configuration transmits test packets with 3 dB gain, which makes the detection of weaker signals easier since it doubles the power.

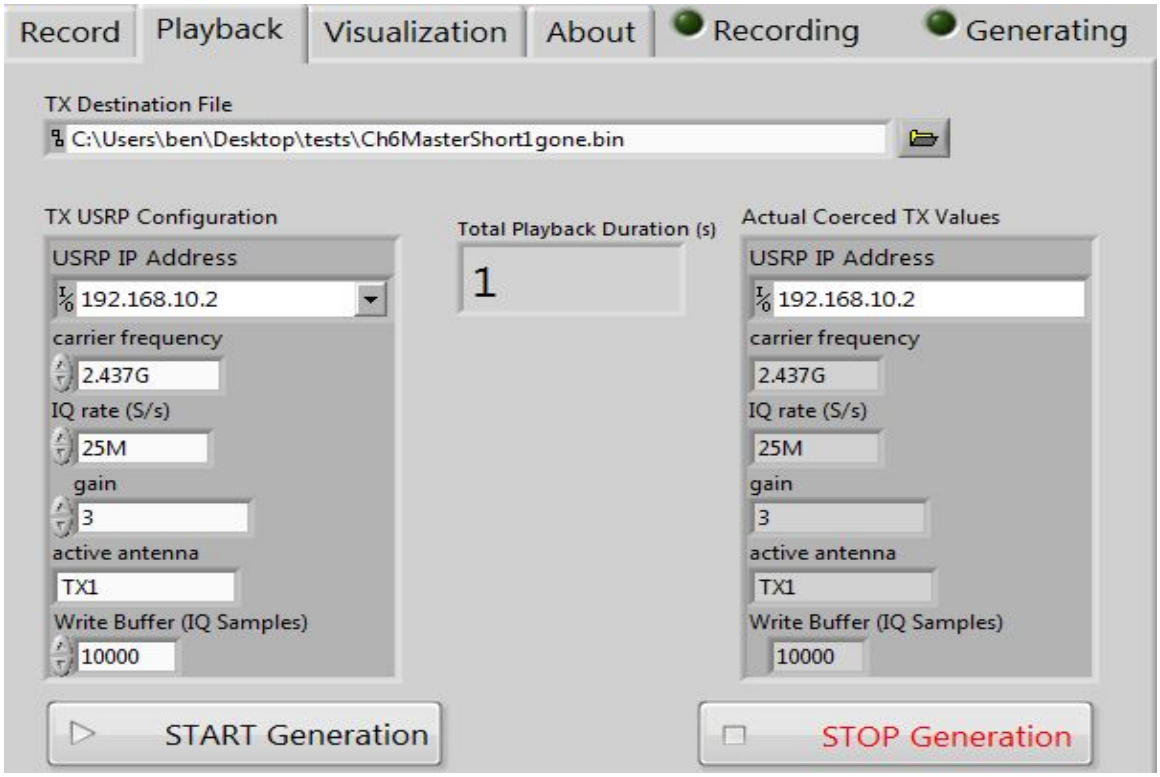

Figure 3.9: USRP Configuration

Binomial confidence intervals of the packet response rates indicate the performance of each transceiver for the designated trial. Confidence intervals are derived from the formula:

$$\hat{p} \pm z_1 - \frac{\alpha}{2}\sqrt{\frac{\hat{p}\,(1-\hat{p})}{n}} \tag{3.1}$$

$\hat{p}$ is the probability of packet response. "z_1", taken at the 99% confidence interval, is 2.57. "$n = 40$", which indicates the number of trials for each experiment. Since each trial

executes 40 iterations of the modified IMCP packets, the central limit theorem has enough data to illustrate a normal distribution. Section 4.3 further analyzes the results of individual packet response rates.

3.5 Methodology Summary

This chapter covers the methodology used to discern differences in packet response as a result of preamble modifications in 802.11b against a series of transceivers. The Ettus USRP–2921 performs the initial capture of wireless traffic. MATLAB processes the raw data capture to create each trial, which consists of the three ICMP packets. The USRP retransmits a burst of ICMP echo requests, with one containing a modified preamble to gauge the probability of packet response. Wireshark verifies connectivity to the Dlink N300 DIR-655 router. Additionally, Wireshark monitors positive acknowledgement of a transceiver to respond to each transmission. Binomial confidence intervals stratify transceiver performance based on packet response.

IV. Results and Analysis

4.1 Introduction

This chapter presents the results and analysis of the data collected during the experiments. Section 4.2 details the Wald–Wolfowitz runs test to validate mutual independence. Section 4.3 discusses the results of packet response from each transceiver. Section 4.4 covers device classification results, Kolmogorov–Smirnov tests and the overall summary of the experimental analysis. Section 4.5 summarizes the results of Chapter 4.

4.2 Validation of Mutual Independence

4.2.1 Wald–Wolfowitz Runs Test.

Before further analyzing the results of data collection, the experiment must pass the test of mutual independence. The Wald–Wolfowitz runs test evaluates the randomness of the data collected to determine mutual independence. This validation is essential to identify factors that otherwise create dependency between each sequence of test packets the transceivers respond to.

In the Wald–Wolfowitz runs test, the null hypothesis asserts that the two-valued data set is randomly distributed and independent. This non-parametric statistical check analyzes a run, which is defined as a sequence of identical values. In the context of SXS, identical values refers to consecutive positive or negative responses to the modified ICMP packet transmitted throughout the experiment. These values comprise a 40–element vector that the Wald–Wolfowitz runs test assesses.

Using statistical analysis, the normal approximation produces the z–value, which obtains the corresponding two–tailed P–value. The Wald–Wolfowitz runs test also accounts for the variable probability of packet response. Trials that contain very high or low response rates produce a smaller number of runs. For example, an Atheros AR928X performs with

a 95% packet response rate out of 40 total transmissions. This indicates that the AR928X transceiver successfully acknowledges 38 out of 40 packets. The maximum number of runs possible given this scenario is five. Conversely, suppose the AR928X performs with a 50% packet response rate during a different trial. The AR928X successfully responds to 20 out of 40 transmissions. The maximum number of runs possible is 40, assuming the AR928X alternates between positive and negative responses. Since the packet response rates amongst the transceivers vary during the experiment, these characteristics favor conducting non–parametric tests through the Wald–Wolfowitz method.

4.2.2 Wald–Wolfowitz Application to SXS System.

As discussed in Section 4.2.1, a series of positive or negative responses equates to a run. For example, suppose a fair coin is flipped 10 times, producing the following results of "tails" (+): $+ + + - - + - + + -$. n_1 represents the number of tails and n_0 represents the number of heads. n equals 10, indicating the number of coin flips. The expected number of runs is:

$$E(R) = 1 + \frac{2n_1 n_0}{n} \;\; or \;\; E(R) = 1 + \frac{2(6)(4)}{(10)} \;\; or \;\; E(R) = 5.8 \tag{4.1}$$

Preliminary analysis concludes that $E(R) = 5.8$, which is very close to the actual observed number of runs – in this example that equals 6. The Wald-Wolfowitz test subsequently accounts for variance (σ^2) that occurs around the expected number of runs. The variance is:

$$\sigma^2 = \frac{2n_0 n_1 \left(2n_0 n_1 - n\right)}{n^2 (n-1)} \tag{4.2}$$

The z–value used to find the corresponding two-tailed P-value requires the observed number of positive and negative runs as well as the number of trials (n). Combined with (4.1) and (4.2), this data produces the expected and observed number of runs and the variance (σ^2). The square root of the variance divides the difference between the observed and expected number of runs, producing:

$$Z = \frac{R - E(R)}{\sigma} \tag{4.3}$$

Using actual observed data from the experiment, the Intel 6250 transceiver produces 13 runs in the first trial. Appendix H contains the values of the packet response rates for trial 1. Testing for mutual independence requires the number of trials and the total for each response (success/failure). Applying (4.1), (4.2) and (4.3) to the Intel 6250, the transceiver responded successfully 33 out of 40 total packet transmissions (82.5%), therefore $n_1 = 33$, $n_0 = 7$, $n = 40$. The expected runs calculates as:

$$E(R) = 1 + \frac{(2)(33)(7)}{40} = 12.55 \tag{4.4}$$

The actual number of observed runs was 13, which suggests mutual independence given the small difference between the expected and observed runs. Calculation of the variance taken from (4.2) yields:

$$\sigma^2 = \frac{((2)(7)(33)((2)(7)(33) - 40))}{\left(40^2(40-1)\right)} = \frac{(462(422))}{(1600(39))} = 3.1244 \tag{4.5}$$

The z–score is derived from (4.3) as:

$$Z = \frac{13 - 12.55}{\sqrt{3.1244}} = \frac{0.45}{1.7676} = 0.2545 \tag{4.6}$$

Lastly, the corresponding P-value for a Z–score at 0.2545 from a two-tailed normal approximation is 0.799. This strongly suggests that the data is mutually independent and free of external factors that otherwise influence the results.

4.2.3 Mutual Independence.

Minitab Software produces the calculations for all transceivers to test for mutual independence [Min14]. Tables 4.1, 4.2 and 4.3 display the results for mutual independence testing of the Intel, Broadcom and Atheros transceivers. The trial numbers correspond to how many bits are removed from the preamble, and a "*" indicates that the tested transceiver failed to respond to any of the 40 test packet transmissions, thus no test for mutual independence is necessary.

Table 4.1: Runs Test Calculations for Intel Transceivers (P–Value)

	Trial 1	**Trial 2**	**Trial 3**	**Trial 4**	**Trial 5**	**Trial 6**	**Trial 7**	**Trial 8**	**Trial 9**	**Trial 10**
Intel 3945	0.595	0.343	0.389	0.43	0.981	0.259	0.611	0.684	0.819	*
Intel 4965	0.799	0.684	0.074	0.749	0.369	0.57	0.819	*	*	*
Intel 6205	0.17	0.799	0.576	0.283	0.399	0.35	0.702	*	*	*
Intel 6250	0.799	0.606	0.684	0.343	0.659	0.947	0.455	*	*	*

Table 4.2: Runs Test Calculations for Broadcom Transceivers (P–Value)

	Trial 1	**Trial 2**	**Trial 3**	**Trial 4**	**Trial 5**	**Trial 6**	**Trial 7**	**Trial 8**	**Trial 9**	**Trial 10**
Broadcom 4311	0.799	1.000	0.455	0.944	0.701	0.341	1.000	0.667	0.646	0.799
Broadcom 4313	0.646	0.944	0.883	0.49	0.341	0.285	0.196	0.759	0.323	0.646

Table 4.3: Runs Test Calculations for Atheros Transceivers (P–Value)

	Trial 1	**Trial 2**	**Trial 3**	**Trial 4**	**Trial 5**	**Trial 6**	**Trial 7**	**Trial 8**	**Trial 9**	**Trial 10**
Atheros 928X	0.576	0.819	0.819	*	*	*	*	*	*	*
Atheros 9001U	0.819	0.897	0.799	*	*	*	*	*	*	*

In summary, all tested transceivers pass the Wald–Wolfowitz runs test at a 1% significance level to validate mutual independence. This statistical analysis strongly suggests that the results collected from the tested transceivers are random. Section 4.3 analyzes the results of the individual transceivers and provides further observations that indicate unique characteristics from packet response rates.

4.3 Results and Analysis of Individual transceivers

4.3.1 Analysis of Intel–Based Transceivers.

The overall performance of the Intel transceivers suggests a steady response to modified preambles up to four bits removed from the preamble. Overall, Intel transceivers respond successfully 81% of the time during trials with up to four bits removed from the preamble. Packet response declines between four and seven bits removed from the preamble. During this stage, overall packet response with Intel transceivers fall to 67%, then 59% after transmitting ICMP packets with six bits removed from the preamble. The final stage of deteriorating packet response occurs when seven bits are removed from the preamble. Most Intel transceivers fail after the eighth trial, with the exception of the Intel 3965 transceiver. All Intel transceivers fail entirely by the time ten bits are removed from the preamble. The next section reports individual Intel transceiver results.

4.3.2 Results of the Intel 3945 Series transceiver.

This particular transceiver, housed inside a Dell Inspiron E1505 laptop, responded most consistently to the ICMP echo requests throughout the experiment. Figure 4.1 illustrates how the transceiver responded greater than 80% of the time successfully to the modified ICMP packets until 8 bits were removed. Packet response deteriorates after the eighth and ninth bits are removed, falling to zero–barrier on the tenth bit.

The Intel 3945 transceiver differs in performance from the three other Intel–based transceivers, strongly suggesting that intra–manufacturing is possible. Statistical analysis through the Kolmogorov–Smirnov test, covered in Section 4.4.3, confirms that the performance between the Intel 3945 transceiver differs from other Intel–based transceivers.

4.3.3 Results of the Intel 4965 Series transceiver.

Out of the four different transceivers from the Intel family this research analyzes, the Intel 4965 transceiver experiences the fastest rate of deterioration. The transceiver operates within a Toshiba Satellite laptop. As Figure 4.2 displays, the packet response drops with

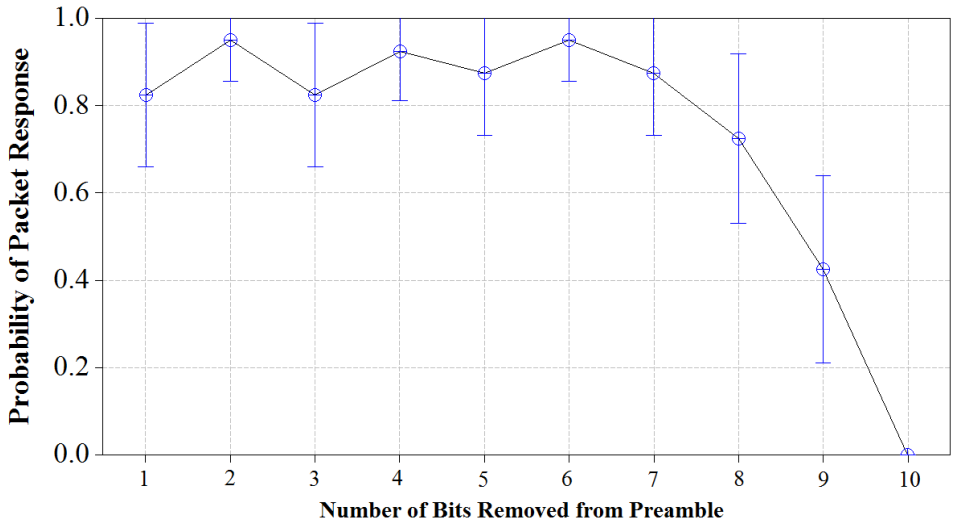

Figure 4.1: Intel 3945 Packet Response Rates to Modified Preambles

three bits removed from the preamble. Packet response rates decline steadily between the third and seventh bits removed, further supporting an inverse relationship between the two factors. The transceiver reaches zero–barrier when eight bits are removed from the preamble.

The Intel 4965 transceiver demonstrates unique characteristics apart from the Intel 3945 transceiver. The Intel 4965 transceiver fails to respond entirely when eight bits are removed from the preamble, whereas the Intel 3945 transceiver continued to respond until ten bits were removed from the preamble. However, the Intel 4965 transceiver shares the same point of complete failure with the Intel 6200 series transceivers. Comparing when two transceivers fail entire is insufficient to perform device classification, thus alternative methods are required to enumerate the transceivers.

Kolmogorov–Smirnov tests indicate clear separation with respect to transceiver performance between the Intel 4965 and 6200 series, especially when analyzing packet response rates with five and six bits removed. Graph analysis indicates no intersection between the 4965 and 6200 series taken at the 99% confidence interval, with six bits

removed from the preamble. This fact supports device classification to differentiate between these transceivers. Furthermore, this observation relates back to the problem statement that addresses Goal #3: Does there exist enough performance difference to conduct device classification?

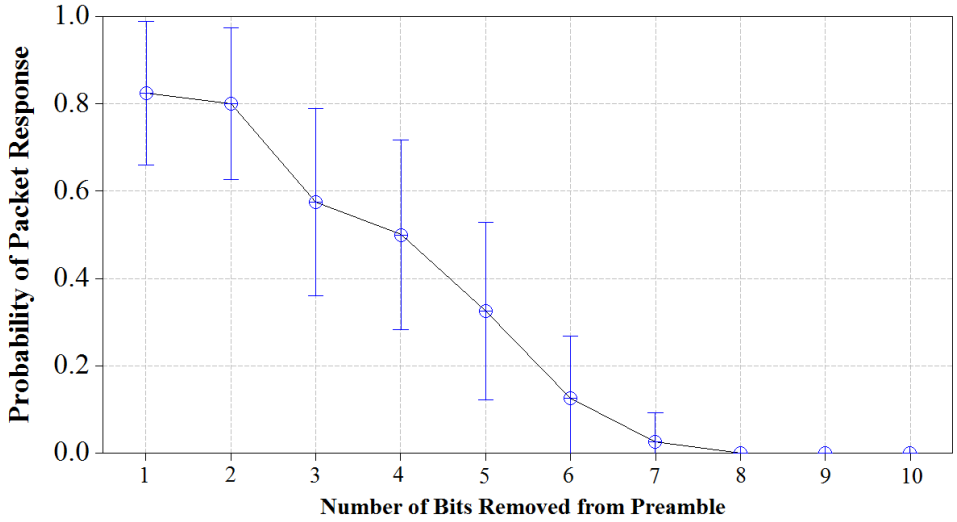

Figure 4.2: Intel 4965 Packet Response Rates to Modified Preambles

4.3.4 Results of the Intel 6250 and Intel 6205 transceivers.

The Intel 6250 transceiver, a more recent wireless adapter released in 2012, provides wireless communications to a Dell Latitude E6520 series laptop. The behavioral properties of the transceiver parallel packet response rates of the Intel Advanced-N 6205 wireless adapter. The Intel Advanced-N 6205 resides inside an HP EliteBook 8570 series laptop. Both transceivers experience an 80% packet response rate throughout the experiment until the initial stages of decline when the fifth bit is removed.

Figure 4.3 displays the performance plots of the Intel 6200 series transceivers. The most subtle differences in packet response with the transceivers occur between the fifth and sixth bits removed from the preamble. The Intel Advanced-N 6205 transceivers plateaus

briefly when the fifth and sixth bits are removed, holding steady at a reasonable 70% response rate. However, the Intel 6250 exhibits a more gradual declination with response rates at 77% and 60% respectively. When executing the seventh trial, packet response rates drop significantly. The greatest separation in performance between the two transceivers occurs during the third trial.

Figure 4.3: Comparison of the Intel 6250 vs. 6205 (99% CI)

4.3.5 Analysis of Atheros–Based Transceivers.

SXS tests two different Atheros transceivers in this experiment. Both transceivers experience similar rates of packet response, despite unique physical implementations. Atheros–based transceivers contain the highest observable sensitivity to packet manipulation. Moreover, the Atheros transceiver demonstrates no gradual decline in packet response when bits

from the preamble are removed. The AR928X and AR9001U transceivers exhibit an abrupt drop after the third and fourth bits removed from the preamble.

4.3.6 Results of Atheros AR928X Series Transceiver.

The Atheros AR928X is an internal wireless adapter operating on a Dell XPS laptop. Out of all the tested transceivers, the AR928X transceiver contains some of the highest response rates when three or less bits are removed. The most notable feature Figure 4.4 illustrates is the abrupt failure that occurs when the fourth bit is removed. During the trial with four bits removed from the preamble, the AR928X sent periodic ICMP echo requests to verify connectivity to the AP due to the unexpected drop in packet response. These results strongly differentiate from Intel transceivers that continue to respond well beyond four bits removed from the preamble.

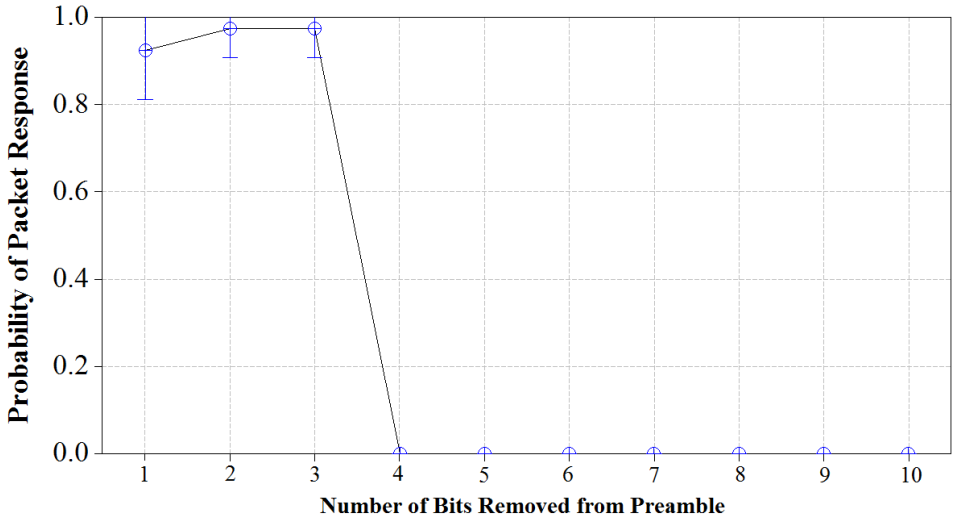

Figure 4.4: Atheros AR928X Packet Response Rates to Modified Preambles

4.3.7 Results of Atheros 9001U-2NX Series Transceiver.

The Atheros 9001U-2NX transceiver “AirPcap”, operates as a USB wireless adapter. Aside from obvious physical differences, AirPcap differs by collecting all RF signals

instead of the specific channel associated to the transceiver under test. Despite collecting traffic on all channels with monitor mode, the AirPcap performs similar to the AR928X series. Figure 4.5 demonstrates a similar performance trend of the AR928X transceiver. Removing one or two bits from the preamble continues to yield high packet response rates. However, removing the third bit causes a drastic drop, followed by complete failure by the fourth removed bit from the preamble.

The Atheros transceivers contain very limited data to perform device classification, due to their sensitivity. Fortunately, despite failing during the same trial, the AR928X and AR9001U transceivers performed very dissimilarly. Where the AR928X transceiver holds steady at a 98% response rate, the AR9001U drops to 18%. This observation confirms the possibility of intra–manufacturer transceiver profiling. Section 4.5 further addresses aggregate device classification.

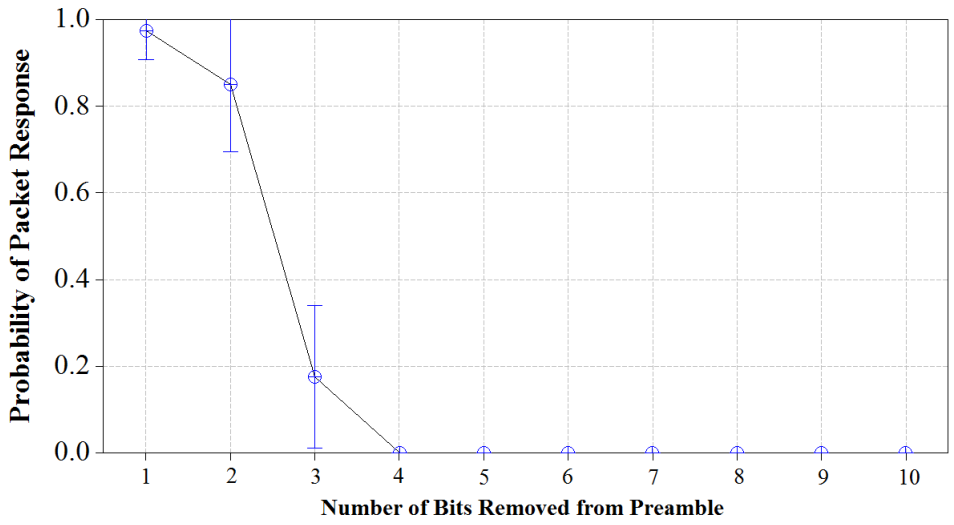

Figure 4.5: Atheros AR9001U-2NX Packet Response Rates to Modified Preambles

4.3.8 Analysis of Broadcom–Based Transceivers.

Between all transceiver SXS tests, Broadcom transceivers perform the most consistently during all trials. Their performance differs substantially from Intel and Atheros transceivers. Both transceivers continue to successfully respond to modified packets even with 10 bits removed from the preamble. However, the transceivers perform analogous to the Intel 6200 with respect to similarities. Despite the inability to disambiguate between the Broadcom transceivers, the packet response rates demonstrate consistency between the transceivers since they are nearly identical in physical implementation.

4.3.9 Results of the Broadcom 4311 and 4313 Transceivers.

In the Intel and Atheros–based transceivers, device classification analyzes when the wireless adapter fails to respond to any of the 40 packets transmitted. Devices that fail at the same point during trials require further analysis of packet response prior to total failure, as the Atheros transceivers demonstrates. However the challenge with both Broadcom–based transceivers stems from the fact that even with 10 bits removed from the preamble, the transceiver continues to successfully respond to test packets. SXS needs to remove more bits in order to obtain the zero–barrier for the Broadcom transceivers.

The Broadcom 4311 averages a 77% response rate overall during the trials. Figure 4.6 compares the results of the two Broadcom–based transceivers. An initial decline occurs between the third and fourth bits removed. However, the Broadcom 4311 transceiver recovers, hovering just under 80% between the fifth and ninth bits removed from the preamble. This response pattern also occurs within the Broadcom 4313 transceiver. Initial trials between the first and third bits removed result in a slight decline with the Broadcom 4313 transceiver. Similarly, the Broadcom 4313 recovers when the fourth and fifth bits are removed. Although this particular transceiver maintains packet response rates between 60% and 80%, the overall average response rate of 68% is lower than observed with the Broadcom 4311 transceiver.

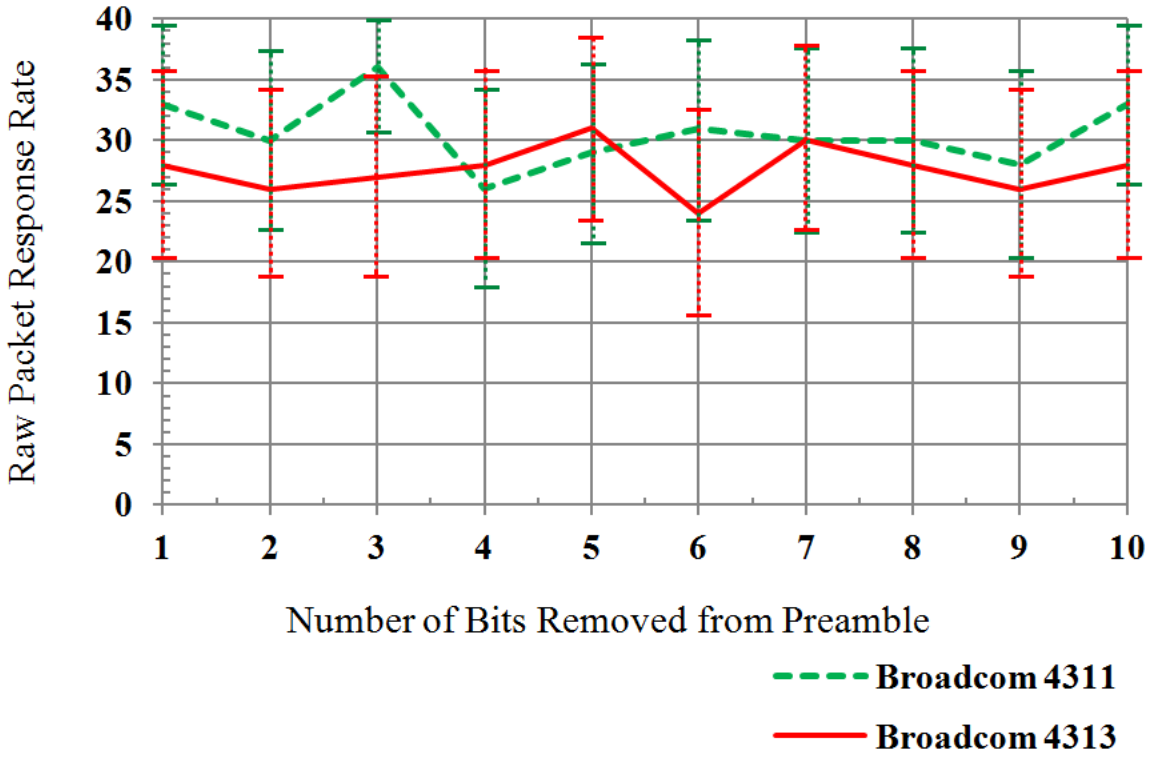

Figure 4.6: Comparison of the Broadcom 4311 vs. 4313 (99% CI)

4.4 Device Classification Results

4.4.1 Analyzing Failure Rates.

There exist various thresholds throughout the experiment when transceivers fail to respond to any test packets, listed in Table 4.4. These thresholds are valuable because they demonstrate that the transceivers under test do not perform in identical fashion. However, only 10 different trials enumerate the transceivers. In spite of the fact that SXS tests eight different transceivers, the fact remains that several transceivers fail during the same trial. Moreover, in the case of the Broadcom transceivers, both transceivers continue to transmit despite 10 bits missing from the preamble.

Three out of the four Intel–based transceivers fail entirely when eight bits are removed from the preamble. The Intel 3945 transceiver successfully responds until ten bits are

Table 4.4: Experimental Trials Identifying "Zero–Barrier"

	Packet Response Rate (Percentages)									
	Trial 1	**Trial 2**	**Trial 3**	**Trial 4**	**Trial 5**	**Trial 6**	**Trial 7**	**Trial 8**	**Trial 9**	**Trial 10**
Atheros 928X	92.5%	97.5%	97.5%	**0.0%**	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Atheros 9001U	97.5%	85.0%	17.5%	**0.0%**	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Broadcom 4311	82.5%	75.0%	90.0%	65.0%	72.5%	77.5%	75.0%	75.0%	70.0%	82.5%
Broadcom 4313	70.0%	65.0%	67.5%	70.0%	77.5%	60.0%	75.0%	70.0%	65.0%	70.0%
Intel 6205	77.5%	82.5%	92.5%	92.5%	70.0%	70.0%	7.5%	**0.0%**	0.0%	0.0%
Intel 6250	82.5%	85.0%	80.0%	87.5%	77.5%	60.0%	10.0%	**0.0%**	0.0%	0.0%
Intel 3945	82.5%	95.0%	82.5%	92.5%	87.5%	95.0%	87.5%	72.5%	42.5%	0.0%
Intel 4965	82.5%	80.0%	57.5%	50.0%	32.5%	12.5%	2.5%	**0.0%**	0.0%	0.0%

removed from the preamble. If a software defined radio utilizes the trial with eight bits removed to enumerate an unknown transceiver, neither the Intel 6200 series nor the Intel 4965 transceiver respond. Assuming Intel represents the total pool of potential transceivers, any positive response strongly suggests the unknown device belongs to an Intel 3945 transceiver.

4.4.2 Device Classification through Trial Analysis.

Analyzing when transceivers fail entirely results in reasonably adequate profiling amongst manufacturers, but falls short when enumerating specific models. In order to successfully classify device specific models, statistical analysis compares packet response rates of the transceivers under test. When computing the expected number of runs to determine mutual independence, the formula factors in the probability of packet response. Similarly, trial analysis factors in circumstances where two different transceivers reach zero–barrier at the same time.

Initial trial analysis compares any two transceivers performance for the duration of the entire experiment. Table 4.5 compares the raw packet response rates between the Intel 3945 and Intel 6205 transceiver. The results of trial analysis produce a numeric vector that represents the absolute difference between two tested transceivers for each individual trial. From this vector, the greatest value identifies the trial where two transceivers differ the most

in packet response. Depending on how great the difference, statistical analysis determines whether or not device classification is possible based on a certain confidence level.

Table 4.5: Trial Analysis: Intel 3945 vs. Intel 6205

	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Trial 6	Trial 7	Trial 8	Trial 9	Trial 10
Intel 3945	33	38	33	37	35	38	**35**	29	17	0
Intel 6205	31	33	37	37	28	28	**3**	0	0	0
Difference	2	5	4	0	7	10	**32**	29	17	0

Conducting trial analysis between the two Intel transceivers illustrates similar performance, until Trial 7. Despite the difference in zero–barrier between these two transceivers, Trial 7 illustrates the greatest performance difference. Kolmogorov–Smirnov tests conclude higher accuracy rates using Trial 7, even though the Intel Advanced-N 6205 elicits three positive responses. Thus, trial analysis ignores when transceivers reach zero–barrier.

Table 4.6 depicts the results of trial analysis between all tested transceivers and all possible trials. As evidenced by the raw data, each cell within Table 4.6 contains the value of greatest separation between each set of transceivers. The intersection between the Intel 3945 and Intel 6205 contains the value 32, which references Trial 7 from Table 4.5. Similarly, all remaining transceivers undergo the same process. This identifies the trial of greatest performance difference and to what degree the two transceivers differ. Kolmogorov–Smirnov tests use these values to determine whether device classification is possible between any two transceivers.

Table 4.6 also illustrates a total of 28 independent comparisons. Most cases contain values around 20 or higher. Two cases failed to meet this criteria. The first circumstance occurs between the Intel 6250 transceiver and Intel Advanced-N 6205 series. The second circumstance occurs between the Broadcom 4313 and Broadcom 4311 series. Both cases involve transceivers not only from the same manufacturer, but also from the same

transceiver family. While device classification is an explicitly stated goal, the observation of similar performance supports the experimental design that validates transceiver response rates.

Table 4.6: Trial Analysis: Packet Response Rate Difference

	Intel 3945	Intel 4965	Intel 6205	Intel 6250	AR928X	AR9001U	BCM4311	BCM4313
Intel 3945		34	32	31	38	38	33	28
Intel 4965			23	19	20	20	33	29
Intel 6205				5	37	37	33	28
Intel 6250					35	35	33	28
AR928X						32	33	30
AR9001U							33	30
BCM4311								9
BCM4313								

When comparing the performance of the Intel 6200 series, the transceivers only differ in response at most by five packets. Similarly, comparing the performance of the Broadcom–based transceivers yields a difference of nine packets. Both circumstances produce the greatest difference during Trial 3. Although trial analysis fails to successfully identify the specific transceiver based on the aforementioned packet response rates, the results eliminate the majority of other potential transceivers. This largely satisfies the device classification goal.

4.4.3 Kolmogorov–Smirnov Test Results.

A two–sample Kolmogorov–Smirnov (K–S) test enables in depth analysis to characterize statistical differences in packet response rates. While trial analysis indicates clear separation in packet response in most cases, K–S tests provide mathematical proof at a certain confidence interval to differentiate between transceivers. The two–sample K–S test compares the empirical distribution function of the two samples and returns a P–value.

The empirical distribution function is:

$$F_n(x) = \frac{1}{n} \sum_{i=1}^{n} I_{X_i} \leq x \tag{4.7}$$

Where $n = 40$ for this research, as it indicates the number of independent observations in a trial. I_{X_i} is the indicator function, which deviates to 1 if $I_{X_i} \leq x$, otherwise 0.

In terms of the K–S test, the null hypothesis asserts that the samples are drawn from the same distribution. The two distributions under test assume equality unless proven otherwise. If the null hypothesis holds true, the two distributions are considered equal, which occurs when device classification fails. Alternatively, if enough evidence exists to reject the null hypothsis, clear separation exists to enumerate the transceiver. Additionally, the K–S demonstrates favorable properties for general applications since the test accounts for differences in the shape of the empirical Cumulative Distribution Function (CDF). The comparison of the empirical CDF between the two samples is:

$$D_{n,n'} = \sup_x \mid F_{1,n}(x) - F_{2,n'}(x) \mid \tag{4.8}$$

Each sample is represented from 4.7 by $F_{1,n}(x)$ and $F_{2,n'}(x)$. $D_{n,n'}$ is the absolute maximal separation, or supremum, of the two samples.

The P–value derives from the comparison of the empirical CDFs between each sample. A low P–value (0.05 or less) strongly suggests that the two transceivers differ. Alternatively, a P–value greater than 0.05 suggests a strong possibility that the two distributions are comparable. Important to note, a P–value greater than 0.05 does *not* mean that the two distributions are the same. It simply means there does not exist enough evidence to suggest otherwise, more commonly referred to in the statistical sense of "not different".

Table 4.7 lists the corresponding P–values for all transceivers under test . Analyzing the Intel 3945 transceiver, results of the K–S tests indicate uniqueness against all other transceivers. The Intel 3945 transceiver results in P–values equating to 5.26E-13 against the Intel 4965 transceiver. The Intel 6200 series performs more closely to the Intel 3945

transceiver, as evidenced by the marginally increased P–values of 1.54E-11 and 7.36E-11. Identical values found in Table 4.7 are general indicators where at least one transceiver fails to respond entirely. The Broadcom Broadcom 4311 exhibits this trait, since Broadcom transceivers continue to successfully acknowledge ICMP packets with 10 bits removed from the preamble. No other transceivers successfully respond during Trial 10 except Broadcom.

Table 4.7: Results of K–S Tests and Corresponding P–values

	Intel 3945	Intel 4965	Intel 6205	Intel 6250	AR928X	AR9001U	BCM4311	BCM4313
Intel 3945		5.62E-13	1.54E-11	7.36E-11	4.44E-16	4.44E-16	3.00E-12	6.15E-09
Intel 4965			3.61E-06	2.41E-04	9.08E-05	9.08E-05	3.00E-12	1.48E-09
Intel 6205				**0.9135**	2.78E-15	2.78E-15	3.00E-12	6.15E-09
Intel 6250					1.00E-13	1.00E-13	3.00E-12	6.15E-09
AR928X						1.52E-11	3.00E-12	7.36E-11
AR9001U							3.00E-12	7.36E-11
BCM4311								**0.2634**
BCM4313								

K–S tests provide supplementary data, when comparing the empirical CDFs between the Intel 6200 series and Broadcom transceivers. In the former, with a maximum packet response difference of five, the Intel 6200 series produced a P–value of 0.9135. This is the highest observed value in the entire experiment. This equates to a maximum performance difference of 12.5%. Figure 4.7 offers an alternative illustration to plot the performance of the two Intel 6200 transceivers. The linear trends mirror one another, in addition to individual transceiver performance during the trials.

Visually, the graph in Figure 4.7 illustrates three key features. Both transceivers exhibit an 84%–86% packet response rate for the first four trials. Initial degradation occurs on the fifth trial, then both transceivers fail when eight bits are removed from the preamble. Linear trends between the two are difficult to disambiguate and further support the findings of the K–S test.

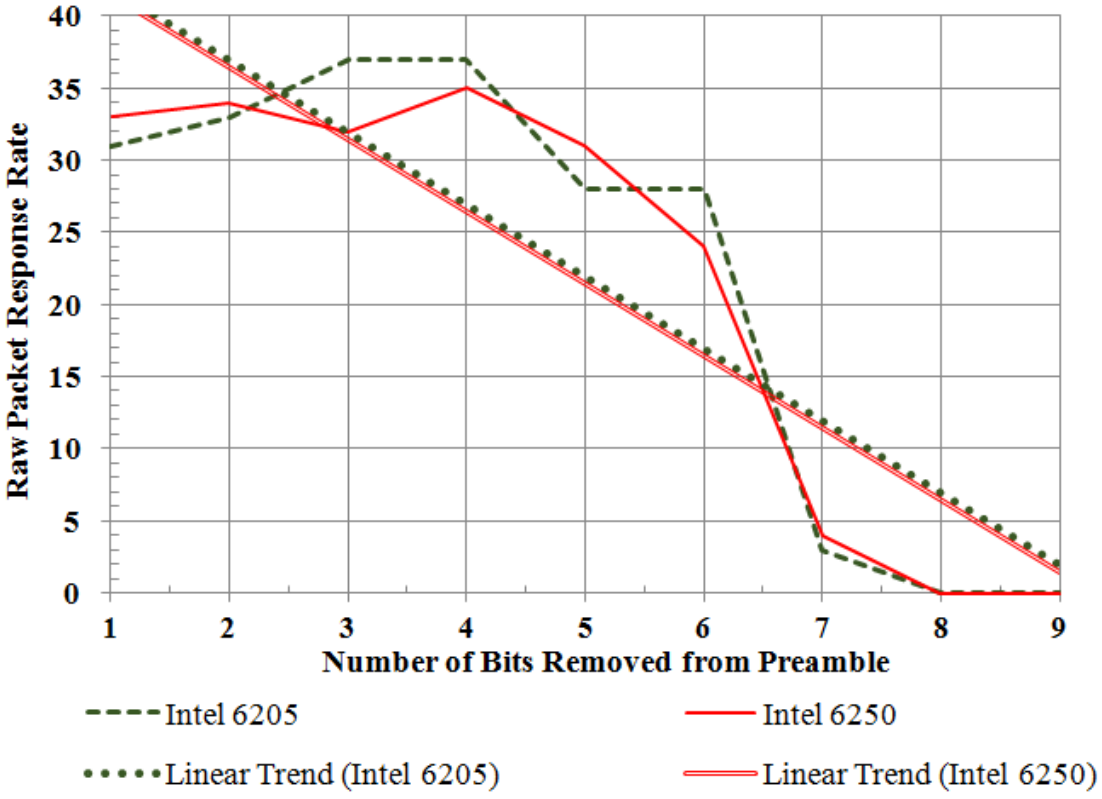

Figure 4.7: Performance and Linear Trend: Intel 6205 vs Intel 6250

The second example where device classification fails occurs when comparing the Broadcom–based transceivers to one another. Similar to the Intel 6200 series transceivers, the greatest separation occurs during the third trial. This observation concerning the third trial is merely coincidental. This does not lend itself to the conclusion that similar performing transceivers differ the most during the third trial. The Broadcom transceivers differ during the third trial by nine successfully received packets, a 22.5% separation. While this trend is almost twice the difference between the Intel 6200 series, the resulting P–value stands at 0.2634, which fails to reject the null hypothesis that the two transceivers are the same.

Table 4.8 illustrates the trials from which performance differs the most. Atheros–based transceivers exemplify considerable performance difference, especially since both

transceivers fail to respond beyond three bits removed from the preamble. The heightened sensitivity to the modified packets results in analysis based more on the transceiver in question than the Atheros transceiver. In other words, since the Atheros transceivers fail early during trials, the amount of data available to compare is limited. As a result, when comparing the Atheros–based transceivers to any other Intel or Broadcom transceiver, the trials where performance differs the most are identical. Table 4.8 illustrates that 6 out of 13 instances of trial analysis involving Atheros transceivers found Trial 4 contains the greatest performance difference.

Conversely, the Broadcom transceivers performance suggests higher tolerance to modified packets. This phenomenon also leads to a particular trial that best demonstrates the performance difference between the Broadcom transceivers. As Table 4.8 indicates, 9 out of the 13 trials point to Trial 10 where packet response rates differ the most. These observations help construct the optimized packet taxonomy, covered further in Section 4.5.

Table 4.8: Potential Trials to Develop Packet Taxonomy

	Intel 3945	**Intel 4965**	**Intel 6205**	**Intel 6250**	**AR928X**	**AR9001U**	**BCM4311**	**BCM4313**
Intel 3945		Trial 7	Trial 7	Trial 7	Trial 6	Trial 6	Trial 10	Trial 10
Intel 4965			Trial 6	Trial 6	Trial 4	Trial 4	Trial 10	Trial 7
Intel 6205				Trial 3	Trial 4	Trial 4	Trial 10	Trial 10
Intel 6250					Trial 4	Trial 4	Trial 10	Trial 10
AR928X						Trial 3	Trial 10	Trial 7
AR9001U							Trial 10	Trial 7
BCM4311								Trial 3
BCM4313								

4.4.4 Device Classification Packet Taxonomy.

The results of zero–barrier inspection, trial analysis and K–S tests validate distinct characteristics in 26 out of 28 comparisons between the tested transceivers. Furthermore, the table matrix from Figure 4.16 indicates a series of trials that are useful for constructing the flowchart for device classification. Certain trials prove more useful than others.

Utilizing the most effective trials optimizes the device classification flowchart, minimizing the required number of packets to transmit and test unknown devices.

4.4.5 Monte Carlo Simulations.

An optimized flowchart depends on three key factors. First, utilizing the data illustrated in Table 4.8, identify the key trials that provide test environments where the transceivers differ. Second, identify the number of packets to test and branch, while ensuring 99% accuracy or greater. Lastly, in the event of trials where the possible transceivers all result in at least one positive response, a threshold to branch. This numeric value must preserve the integrity of ensuring 99% accuracy. Figure 4.17 lists the cumulative enumeration accuracy rates for all tested transceivers, covered extensively in the next sections.

As shown in Table 4.8, only 5 out of the 10 total trials are necessary to complete the flowchart. Aside from the initial branch of the flowchart (Trial 8), Figure 4.16 references all other trials at some point where transceiver performance differs the greatest. These include: Trial 3, Trial 4, Trial 6 and Trial 10. Depending on the results of transceiver performance during a trial, there exist varying levels of minimum packets to transmit and ensure high fidelity. Lastly, Trial 3 and Trial 6 require a certain threshold established to discern the behavioral pattern most congruent with the correct transceiver.

The formula utilized to obtain the minimum packets to test during select trials is $\left[1 - (1 - p)^n\right] \geq 99\%$; where "p" equals the probability of response and "n" represents the minimum number of test packets. "n" must be sufficiently high enough to maintain 99% accuracy in the worst case scenario. Fortunately, device classification occurs with a minimum of two trials and a maximum of three.

The Atheros transceivers require three total tests. Both transceivers fail entirely with four bits or more removed from the preamble. If an unknown device operates as an Atheros transceiver, the optimized flowchart gradually enumerates the device by reducing the

number of bits removed until the third test (Trial 3). This process systematically eliminates the Broadcom and Intel transceivers as potential transceivers under test in as little as 18 packets. Assuming an Atheros transceiver operating consistently with the findings of this experiment, accuracy rates approach 100% during the first two tests.

Trial 3 introduces the first of two circumstances where positive and negative responses fail the enumeration process. Monte Carlo simulations, computed as $f(x) = \sum_{k=0}^{x} \frac{n!}{k!(n-k)!} p^k (1-p)^{n-k}$ combines with the cumulative binomial distribution probability to provide objective criteria and accurately classify the unknown device. In the case of Trial 3, 15 packets represents the border that separates the performance expected between the AR928X and AR9001U transceivers. The next section illustrates a series of scatterplots and Monte Carlo simulations that explain the math behind the selection of 15 packets to discern between the transceivers.

The Intel 6200 and Intel 4965 transceivers fail to respond in Trial 8, similar to their Atheros counterpart. However, during Trial 4 all three Intel transceivers succeed in responding to at least one packet. The odds of an Intel 6200 series responding at least once successfully during Trial 4 is practically 100%. Using the lesser responsive transceiver (Intel 6250), with a packet response rate of 87.5% during Trial 4, enumeration accuracy rate is $\left[1 - \left(1 - \frac{35}{40}\right)^{10}\right] = \left[1 - \left(9.3x10^{-10}\right)\right]$ or nearly 100%. The Intel 4965 responded less successfully during Trial 4. Subsequently, enumeration accuracy computes to $\left[1 - \left(1 - \frac{20}{40}\right)^{10}\right] = \left[1 - \left(9.7x10^{-4}\right)\right]$ or 99.902%.

The third test finalizes the enumeration process by transmitting 40 packets using Trial 6. This test requires the most packets out of any other, since the Intel 4965 exhibited a very low response rate with six bits removed from the preamble. Again, using the lesser responsive transceiver of the Intel 6200 series, enumeration accuracy stands at $\left[1 - \left(1 - \frac{24}{40}\right)^{40}\right] = \left[1 - \left(1.2x10^{-16}\right)\right]$ or nearly 100%. The accuracy rate to discern an Intel 4965 during Trial 6 is $\left[1 - \left(1 - \frac{5}{40}\right)^{10}\right] = \left[1 - \left(4.8x10^{-3}\right)\right]$ or 99.521%. In the case

of Trial 6, 13 packets represent the border that separates the performance expected between the Intel 6200 series and the Intel 4965 transceiver. The cumulative enumeration accuracy rate for the Intel 6200 series is practically 100%, while the Intel 4965 transceiver equals 99.423%.

Lastly, device classification between the Broadcom 4300 series and Intel 3945 transceivers requires two tests. The cumulative binomial confidence distribution suggests a minimum of 99.993% accuracy for classification under Trial 8. This accuracy rate between the Broadcom 4300 series derives from the least responsive transceiver for the test (Trial 8), the Broadcom 4313 transceiver. This transceiver's packet response rate is 28/40, therefore the formula computes as $\left[1-\left(1-\frac{28}{40}\right)^{8}\right] = \left[1-\left(6.5x10^{-5}\right)\right]$ or 99.993%. The Intel 3945 transceiver performs very similarly at 29/40, therefore the enumeration accuracy rate for Trial 8 computes as $\left[1-\left(1-\frac{29}{40}\right)^{8}\right] = \left[1-\left(3.3x10^{-5}\right)\right]$ or 99.997%.

The accuracy during Trial 10 for an Intel 3945 transceiver is 100%, since the transceiver fails to respond to any of the test packets. Thus, the cumulative accuracy when the transceiver under test is an Intel 3945 equals 99.997%. Similarly, the Broadcom transceiver responds fairly consistently even with 10 bits removed from the preamble. The accuracy rate of the Broadcom 4311, with a packet response rate of 33/40, equals $\left[1-\left(1-\frac{33}{40}\right)^{4}\right] = \left[1-\left(9.4x10^{-4}\right)\right]$ or 99.906%. The BCM 4313 enumeration accuracy computes as $\left[1-\left(1-\frac{28}{40}\right)^{4}\right] = \left[1-\left(8.1x10^{-3}\right)\right]$ or 99.190%. The cumulative accuracy rate for the Broadcom 4311 is 99.904%, while the accuracy rate for the Broadcom 4313 is 99.183%. Meanwhile, the Intel 3945 transceiver cumulative enumeration accuracy rate is 99.997%.

4.4.6 Scatterplot Diagrams: Trial 3, Trial 6.

Five possible trials enumerate a transceiver, three of which branch solely on either positive or negative responses. However, the remaining two trials test a set of transceivers that all elicit a positive response. MATLAB processes a series of Monte Carlo simulations

Table 4.9: Overall Transceiver Accuracy Rates

	Trial 8	Trial 4	Trial 3	Trial 6	Trial 10	Cumulative
AR928X	100.00%	100.00%	100.00%			100.00%
AR9001U	100.00%	100.00%	99.998%			99.998%
Intel 4965	100.00%	99.902%		99.521%		99.423%
Intel 6205	100.00%	100.00%		100.00%		100.00%
Intel 6250	100.00%	100.00%		100.00%		100.00%
Intel 3945	99.997%				100.00%	99.997%
BCM 4311	99.998%				99.906%	99.904%
BCM 4313	99.993%				99.190%	99.183%

to produce scatterplot diagrams. Each plot of data represents a single iteration, for a total of 100 iterations per transceiver. These diagrams offer a visual representation to illustrate where an appropriate boundary exists to separate transceiver performance. Appendix I contains the code for the MATLAB script.

Trial 3 results indicate 100% accuracy to perform device classification when the transceiver under test is an AR928X transceiver. Similarly, the cumulative binomial distribution asserts a 99.998% accuracy rate when the device in question is an AR9001U transceiver. These results meet the $\geq$ 99% criteria. Figure 4.8 displays the scatterplot diagram for Trial 3, which indicates clear separation between the AR928X and AR9001U group. In fact, the odds of conducting Trial 3 and obtaining a packet response rate between 37.5%–62.5% are practically zero, since preceding trials branch in a manner to factor out all other potential transceivers. Figure 4.8 illustrates the results of the scatterplot diagram. Accordingly, 20 packets represents the appropriate threshold that separates the performance of the AR928X and AR9001U transceivers.

As discussed in the previous section, Trial 6 maintains the accuracy rate $\geq$ 99%. Since the pool of potential transceivers all respond at least once during the trial, the branching criteria does not hinge on a positive or negative response. Monte Carlo simulations produce a scatterplot diagram, however the area separating the two groups is substantially smaller.

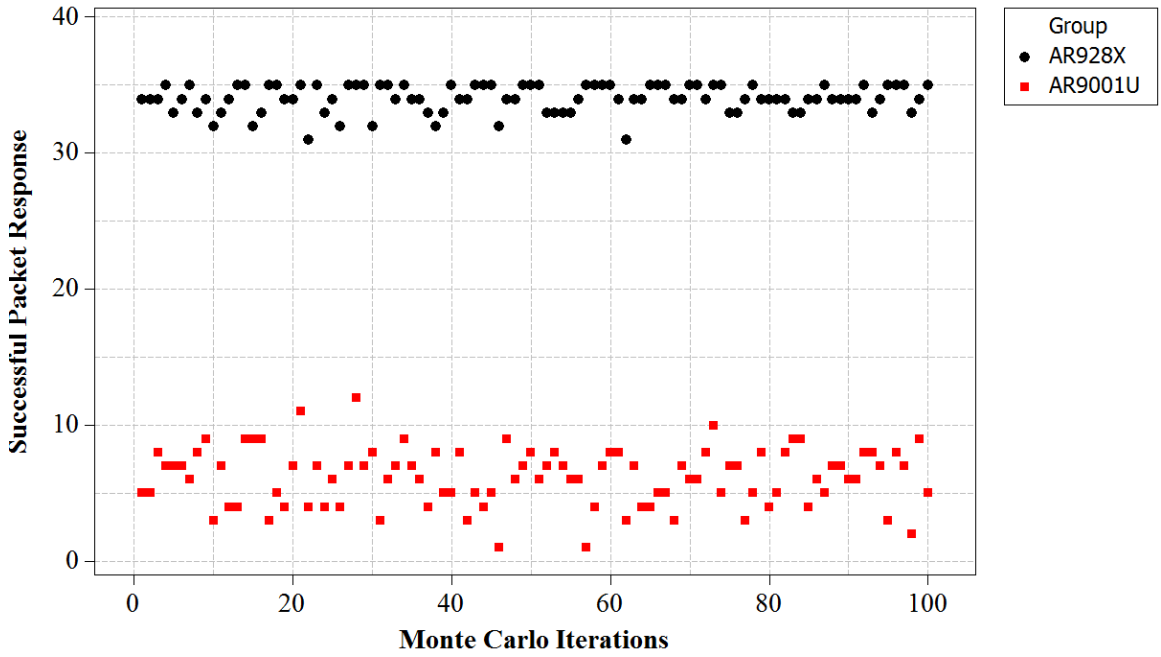

Figure 4.8: Scatterplot Simulation (Trial 3)

Figure 4.9 contains unique properties that illustrate the scatterplot for Trial 6. First, the Intel 6200 series represents the transceiver with a superior packet response rate than the Intel 4965 transceiver. Both groups contain outliers that factor into where best to determine the appropriate threshold. The Intel 4965 simulations contain a single outlier on the 37th iteration, where the simulated packet response rate is 220% greater than observed during the experiment. Similarly, the Intel 6200 series contained an outlier on the 29th iteration, where the simulated packet response rate falls to 37.5%, or 15/40. The observed packet response rate during the experiment holds steady at 60%, or 24/40. Despite the two extremes between the samples, the greatest simulated value of the Intel 4965 transceiver does not breach the lowest simulated value of the Intel 6200 series. Accordingly, 13 packets represents the most appropriate threshold that separates the performance of the Intel 6200 series and Intel 4965 transceivers.

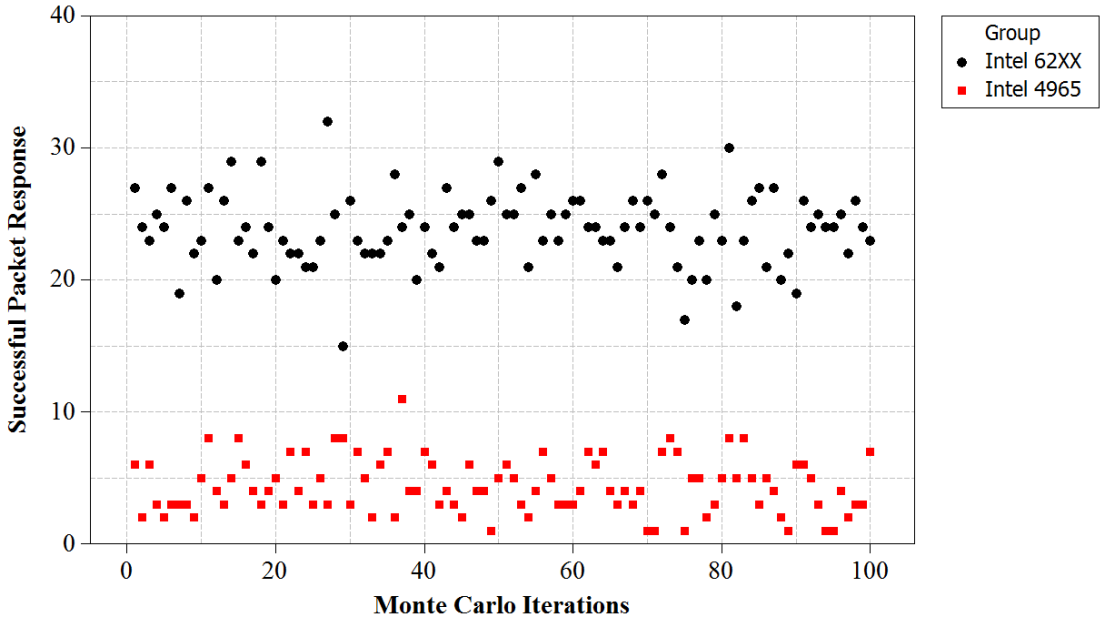

Figure 4.9: Scatterplot Simulation (Trial 6)

4.5 Summary

In summary, five strategically selected trials optimize the packet taxonomy to enumerate the tested transceivers. Highly accurate results with the Intel 3945 and Broadcom transceivers are achievable in as little as 12 packets. All transceivers under test meet cumulative enumeration accuracy rates greater than 99%. Scatterplots and Monte Carlo simulations illustrate appropriate thresholds concerning trials where positive and negative responses fail. Figure 4.10 presents the final packet taxonomy to successfully enumerate all transceivers.

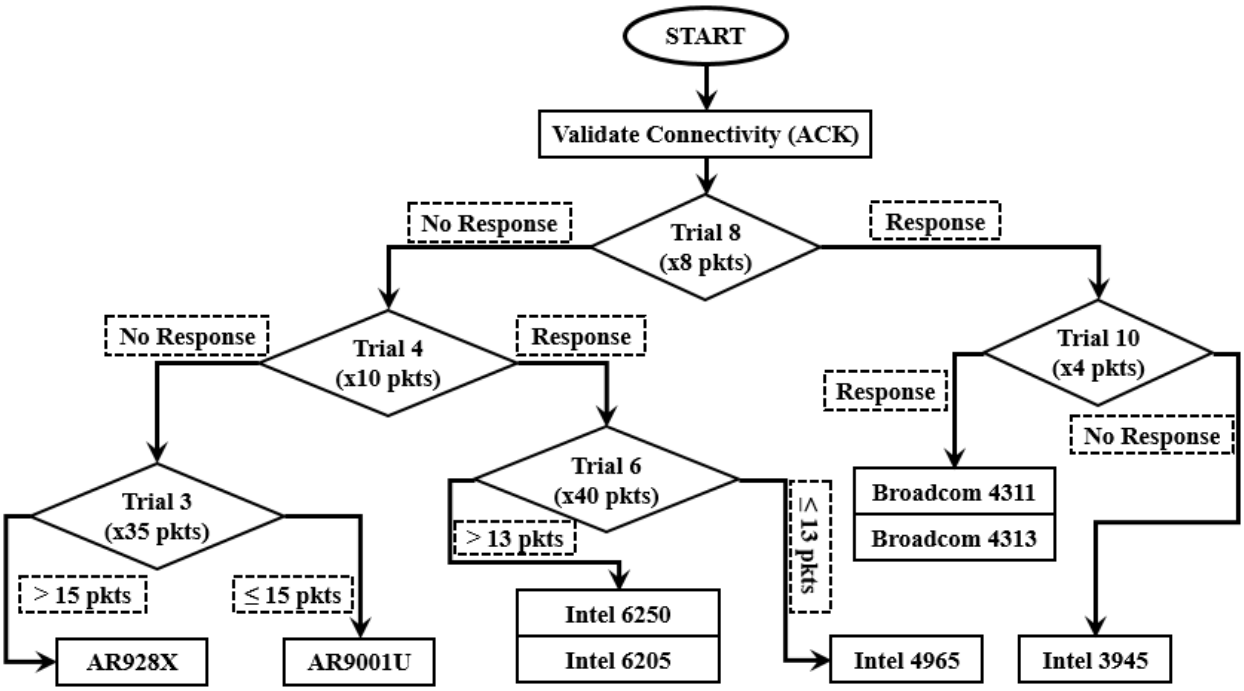

Figure 4.10: Full Device Classification Flowchart

V. Conclusions

5.1 Introduction

This chapter provides the overall conclusions drawn from the research. Section 5.2 discusses whether or not the goals of this research were satisfied. Section 5.3 presents the relevance of this research and impacts on network security. Lastly, Section 5.4 lists areas for future work and potential goals.

5.2 Conclusions of Research

5.2.1 Goal 1: Determine Capability to Replay Modified 802.11b Traffic.

The transceivers under test successfully demonstrate the ability to respond to replay packets transmitted through an arbitrary waveform generator. Software defined radios, such as the NI Ettus USRP–2921, can successfully observe and collect 802.11b wireless traffic. MATLAB stores, handles and modifies the binary vectors representing instantaneous amplitude signatures of the wireless traffic. MATLAB data plots provide visual representation to distinguish between channel noise and wireless traffic. Lastly, MATLAB isolates the data and enables retransmission through the USRP to verify ICMP packet response. Network protocol analyzers such as Wireshark confirm retransmission, as indicated by the sequence number in each successful packet response. The aforementioned validates the physical capability of Goal 1.

5.2.2 Goal 2: Determine Transceivers Performance Differences.

The data collected during the experiment indicates a clear difference in performance between the tested transceivers. The transceivers exhibit varying levels of sensitivity and fail entirely during different trials. Wald–Wolfowitz runs test validate mutual independence of the data and subsequent statistical analysis further supports the evidence to draw causal conclusions. Kolmogorov–Smirnov tests observe behavioral differences between each of

the tested transceivers, with the exception of the two cases discussed in Section 4.4.3. Despite the two isolated circumstances where absolute device classification fails, the experiment successfully enumerates 26 out of 28 observations through trial analysis. This largely satisfies the criteria of Goal 2 and offers valuable observations towards future work the next section covers.

5.2.3 Goal 3: Determine the Optimized Packet Taxonomy.

The optimized flowchart builds upon the conclusions drawn from Goal 2. Key trials provide the branches that enable the classification of devices while minimizing the required number of packets to enumerate the transceiver. Monte Carlo simulations identify the number of packets to transmit during all tests to ensure 99% or better accuracy. Scatterplot diagrams provide visual observations in circumstances where positive and negative responses fail to successfully enumerate an unknown device. In addition, scatterplot diagrams also identify the appropriate threshold to branch during Trial 3 and Trial 6. The cumulative binomial distribution function quantifies the probability of each branch based on the packet response rate of the potential transceivers in question. The results indicate that all transceivers under test satisfy device classification accuracies of 99% or better. Optimizing trial analysis to perform device classification is achievable under two seconds, factoring in the exchange of packets and computing the results of the transceiver under test. Overall, statistical analysis supports high fidelity rates, low packet transmission requirements and satisfies Goal 3.

5.3 Significance of Research

Data emanation, weak encryption and the ability to operate free of cables results in a breeding ground for malicious activity. Current to date, there exist default security practices that fail to harden a wireless access points upon initial configuration. Weak encryption standards, default administrative passwords and poor security practices make cyber attacks easier to execute – costing hundreds of millions annually in intellectual property theft

[Pon12]. Therefore a series of security practices have been developed over the years to provide defense–in–depth.

MITM attacks exist as a result of no authentication requirements upon receipt of an ARP packet. These threats continue to affect LANs when users obtain the ability to arbitrarily assign their MAC address to the communicating interface, such as their wireless adapter. As a result, related research efforts have put forth unique methodologies to introduce stronger security practices to combat MAC spoofing.

RSSI–based models analyze the received signal strength indicator for a specific transceiver. The mesh approach to triangulate an accurate proximity to the transceiver fails when operating in a mobile network. Subsequently, sequence number rate analysis (SNRA) methodologies attempt to address the mobility issue by employing a series of heuristics to discern subtle gaps in sequence numbers. These gaps result when an attacker incorrectly guesses the next sequence number, triggering a possible alert. However, SNRA models fail to address circumstances when a client moves out of range of the network and returns at a later time. SNRA techniques compromise the integrity of what the model is intended to prevent by making exceptions, such as the wrap–around effect when the sequence resets from 4095 to 0. Lastly, OUI prefix validation provides minimal security against users that change their MAC address without any regard to the validity of the prefix. Prefix validation is easily defeated by using legitimate MAC addresses.

This research differs from RSSI, SNRA and prefix validation models. This RF fingerprinting techniques embrace mobile networking by exploiting subtle design differences that lead to differing packet response rates. Preamble modification offers an additional layer to the defense–in–depth of LANs by reducing the surface area an attacker exploits. The significance of this research provides alternative network defense options to combat MAC spoofing. The act of changing unique digital identifiers strongly suggests potential malicious activity due to MITM, ARP cache and DNS poisoning attacks that

follow. SXS detects deviations from the expected packet response rates in a given network environment, limiting or denying access to the rogue transceiver. Automating the detection of potentially malicious devices and restricting access to the network until transceiver verification creates significant barriers that thwart would–be attackers. Even circumstances where device classification is not absolute, the exclusion of other potential transceiver types reduces the attack surface to masquerade as a legitimate user.

5.4 Recommendations for Future Research

This research extends the efforts put forth to produce an RF fingerprinting technique in the IEEE 802.15.4 low–rate wireless Personal Area Network (PAN). The methodology in 802.11b strongly suggests the potential for future research in three key areas. First, addressed in Section 2.5.3, 802.11b utilizes standard long preambles. Future research into other protocols that utilize short preambles may yield evidence that illustrates transceiver differences based on preamble modifications. In addition, 802.11b utilizes DSSS. Future research efforts may find behavioral differences when conducting a similar methodology in OFDM or FSSS–based modulation implementations. Lastly, 802.11b came to fruition in 1999. Most commercial hardware is backwards compatible with this legacy protocol. This experiment demonstrated successful results, validating the RF fingerprinting technique. Future work may also include testing this same methodology against more current protocols, such as IEEE 802.11g, 802.11n, 802.11ac or even 802.11ad.

Exploring future research in RF fingerprinting techniques offers enhanced device classification capabilities. There exists the potential that two similarly performing transceivers operate differently, when analyzed in a different context. This suggests the possibility of expanding preamble modifications through the use of multiple software defined radios by layering the experiments in sequence. As an example, a sequence of experiments may analyze wireless protocols in order: 802.11a, 802.11g, 802.11n. Alternatively, sequential ordering may incorporate modulation schemes such as: DSSS,

FSSS, OFDM. Lastly, protocols that operate with the functionality of using short and long preambles increase the factor by two.

5.5 Summary

Chapter 5 presents the conclusions with recommendations concerning future research. In addition, this chapter contrasts the SXS methodology against other related research efforts. Lastly, this chapter also addresses the success of each stated goal and discusses the relevance of this research with respect to network security.

Appendix A: Performance of Atheros AR9001-U2NX (AirPcap)

	Atheros AR9001U-2NX (AirPcap)									
	Dell Precision 4500 Series									
	1 denotes positive response; 0 denotes negative response									
	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
1	1	1	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0
3	1	1	1	0	0	0	0	0	0	0
4	1	1	0	0	0	0	0	0	0	0
5	1	1	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0
7	1	0	1	0	0	0	0	0	0	0
8	1	1	1	0	0	0	0	0	0	0
9	1	1	0	0	0	0	0	0	0	0
10	1	1	0	0	0	0	0	0	0	0
11	1	1	0	0	0	0	0	0	0	0
12	1	1	0	0	0	0	0	0	0	0
13	1	1	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0
15	1	1	0	0	0	0	0	0	0	0
16	1	1	0	0	0	0	0	0	0	0
17	1	1	1	0	0	0	0	0	0	0
18	1	1	0	0	0	0	0	0	0	0
19	1	1	0	0	0	0	0	0	0	0
20	1	0	0	0	0	0	0	0	0	0
21	1	1	0	0	0	0	0	0	0	0
22	1	1	0	0	0	0	0	0	0	0
23	1	0	0	0	0	0	0	0	0	0
24	1	1	0	0	0	0	0	0	0	0
25	1	1	1	0	0	0	0	0	0	0
26	1	1	0	0	0	0	0	0	0	0
27	1	1	1	0	0	0	0	0	0	0
28	1	1	0	0	0	0	0	0	0	0
29	1	1	0	0	0	0	0	0	0	0
30	1	1	0	0	0	0	0	0	0	0
31	1	0	0	0	0	0	0	0	0	0
32	1	1	0	0	0	0	0	0	0	0
33	1	1	0	0	0	0	0	0	0	0
34	1	1	1	0	0	0	0	0	0	0
35	1	1	0	0	0	0	0	0	0	0
36	1	1	0	0	0	0	0	0	0	0
37	1	1	0	0	0	0	0	0	0	0
38	1	1	0	0	0	0	0	0	0	0
39	0	1	0	0	0	0	0	0	0	0
40	1	1	0	0	0	0	0	0	0	0

Figure A.1: Observed Raw Data: Atheros AR9001U-2NX (AirPcap)

Appendix B: Performance of Atheros AR928X

	Atheros AR928X									
	Dell XPS Series									
	1 denotes positive response; 0 denotes negative response									
	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
1	1	1	1	0	0	0	0	0	0	0
2	1	1	1	0	0	0	0	0	0	0
3	0	1	1	0	0	0	0	0	0	0
4	1	1	1	0	0	0	0	0	0	0
5	1	1	1	0	0	0	0	0	0	0
6	1	1	1	0	0	0	0	0	0	0
7	1	1	1	0	0	0	0	0	0	0
8	1	1	1	0	0	0	0	0	0	0
9	1	1	1	0	0	0	0	0	0	0
10	1	1	1	0	0	0	0	0	0	0
11	1	1	1	0	0	0	0	0	0	0
12	1	1	1	0	0	0	0	0	0	0
13	1	1	0	0	0	0	0	0	0	0
14	0	1	1	0	0	0	0	0	0	0
15	1	1	1	0	0	0	0	0	0	0
16	1	1	1	0	0	0	0	0	0	0
17	1	1	1	0	0	0	0	0	0	0
18	1	1	1	0	0	0	0	0	0	0
19	1	0	1	0	0	0	0	0	0	0
20	1	1	1	0	0	0	0	0	0	0
21	1	1	1	0	0	0	0	0	0	0
22	1	1	1	0	0	0	0	0	0	0
23	1	1	1	0	0	0	0	0	0	0
24	1	1	1	0	0	0	0	0	0	0
25	1	1	1	0	0	0	0	0	0	0
26	1	1	1	0	0	0	0	0	0	0
27	0	1	1	0	0	0	0	0	0	0
28	1	1	1	0	0	0	0	0	0	0
29	1	1	1	0	0	0	0	0	0	0
30	1	1	1	0	0	0	0	0	0	0
31	1	1	1	0	0	0	0	0	0	0
32	1	1	1	0	0	0	0	0	0	0
33	1	1	1	0	0	0	0	0	0	0
34	1	1	1	0	0	0	0	0	0	0
35	1	1	1	0	0	0	0	0	0	0
36	1	1	1	0	0	0	0	0	0	0
37	1	1	1	0	0	0	0	0	0	0
38	1	1	1	0	0	0	0	0	0	0
39	1	1	1	0	0	0	0	0	0	0
40	1	1	1	0	0	0	0	0	0	0

Figure B.1: Observed Raw Data: Atheros AR928X

Appendix C: Performance of Broadcom BCM 4311

	Broadcom BCM 4311									
	Dell XPS Series									
	1 denotes positive response; 0 denotes negative response									
	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
1	1	1	1	0	1	1	0	1	1	1
2	1	0	1	1	1	1	1	0	1	1
3	1	1	1	1	1	0	1	1	0	1
4	0	1	1	1	1	1	1	1	0	1
5	1	1	1	0	0	1	1	1	1	1
6	1	1	1	0	1	1	1	0	1	1
7	1	1	0	1	1	0	1	0	1	0
8	1	1	1	1	1	0	0	1	1	0
9	0	1	1	1	1	1	0	1	0	1
10	0	1	1	1	0	1	1	1	1	1
11	1	0	1	0	0	1	1	1	1	0
12	1	0	1	1	1	0	1	1	0	1
13	1	1	1	1	1	1	0	1	1	1
14	1	1	0	0	1	0	1	1	1	0
15	1	0	1	1	0	1	1	1	1	1
16	1	1	1	1	1	0	1	0	1	1
17	0	1	1	1	1	1	1	1	0	1
18	1	1	1	1	1	1	1	1	0	1
19	1	1	1	0	0	1	1	1	1	0
20	1	1	1	1	1	1	0	1	0	1
21	1	0	1	1	1	1	1	1	0	1
22	1	1	1	0	1	1	1	0	1	1
23	1	1	1	0	1	1	0	1	1	0
24	1	0	1	0	1	1	1	0	1	1
25	1	1	1	1	1	1	0	1	1	1
26	1	1	1	1	1	1	1	0	0	1
27	1	1	1	1	0	1	1	1	1	1
28	0	1	0	1	0	0	1	1	1	0
29	1	1	1	0	1	1	0	1	1	1
30	1	0	0	0	1	1	0	1	1	1
31	1	0	1	1	1	1	1	1	0	1
32	1	1	1	1	1	1	1	0	1	1
33	1	0	1	1	0	1	0	1	1	1
34	0	1	1	0	1	1	1	1	1	1
35	1	1	1	1	1	0	1	1	0	1
36	1	1	1	0	0	1	1	1	1	1
37	1	1	1	1	1	1	1	0	0	1
38	0	1	1	1	1	1	1	0	1	1
39	1	1	1	1	0	0	1	1	1	1
40	1	0	1	0	0	1	1	1	1	1

Figure C.1: Observed Raw Data: Broadcom BCM4311

Appendix D: Performance of Broadcom BCM 4313

	Broadcom BCM 4313									
	Dell Precision 4500 Series									
	1 denotes positive response; 0 denotes negative response									
	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
1	1	1	1	0	1	1	1	1	0	1
2	1	1	1	1	1	0	0	1	1	1
3	1	1	1	1	1	1	1	0	1	1
4	1	0	1	0	0	0	1	1	0	0
5	1	0	1	1	1	0	1	0	0	1
6	1	0	1	1	1	0	1	1	1	0
7	0	1	0	1	1	1	0	1	1	0
8	1	0	1	0	1	1	1	1	1	1
9	0	1	1	0	1	1	1	1	1	1
10	1	0	1	1	1	1	1	0	0	1
11	1	1	1	0	1	1	0	1	0	1
12	1	1	0	0	1	0	0	0	1	1
13	0	1	1	1	0	1	1	0	1	0
14	0	1	0	1	1	0	1	1	0	1
15	1	1	1	1	0	0	1	1	1	1
16	1	0	1	0	1	0	0	1	1	1
17	1	1	0	0	1	0	1	1	1	0
18	1	0	0	1	0	0	1	0	1	1
19	0	0	1	1	1	1	1	0	0	0
20	1	0	1	1	1	1	1	1	1	1
21	1	1	0	1	0	1	1	1	1	0
22	0	0	1	0	1	0	1	1	1	0
23	1	1	1	1	0	0	0	1	1	1
24	1	1	1	1	1	1	1	1	0	1
25	0	1	0	1	1	1	0	1	1	0
26	0	1	1	1	1	1	1	0	1	1
27	1	1	0	1	1	1	1	1	1	1
28	1	1	0	1	0	1	1	1	0	1
29	0	1	1	1	1	1	0	0	1	1
30	1	0	1	1	1	1	1	1	0	1
31	1	0	1	1	1	1	1	1	1	1
32	0	1	1	1	1	1	1	1	0	1
33	0	1	0	0	1	0	0	1	0	1
34	1	0	1	0	1	0	1	1	1	1
35	1	1	1	1	1	1	0	1	0	1
36	1	1	0	1	0	0	1	1	1	0
37	1	0	0	1	0	1	1	0	0	1
38	0	1	1	1	1	1	1	0	1	0
39	1	1	1	0	1	0	1	0	1	0
40	1	1	0	1	1	1	1	1	1	1

Figure D.1: Observed Raw Data: Broadcom BCM 4313

Appendix E: Performance of Intel 3945

	Intel 3945									
	Dell Inspiron E1505									
	1 denotes positive response; 0 denotes negative response									
	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
1	1	1	1	1	0	1	1	1	0	0
2	1	1	1	1	1	0	0	0	0	0
3	1	1	0	1	1	1	1	1	1	0
4	1	1	1	1	1	1	1	1	0	0
5	1	1	1	1	1	1	1	1	1	0
6	1	1	1	1	1	1	1	1	0	0
7	1	1	0	1	1	1	1	1	1	0
8	1	0	1	1	1	1	1	1	1	0
9	1	1	1	1	1	1	1	0	1	0
10	1	1	0	1	1	1	1	1	0	0
11	0	1	1	1	1	1	1	0	0	0
12	1	1	1	1	1	1	0	1	0	0
13	1	0	1	1	1	1	1	0	1	0
14	1	1	0	1	0	1	1	1	1	0
15	1	1	1	1	1	1	1	1	0	0
16	1	1	1	1	1	1	1	0	1	0
17	0	1	1	1	1	1	1	1	1	0
18	0	1	1	0	1	1	1	1	1	0
19	1	1	1	1	1	1	1	1	0	0
20	1	1	1	1	1	1	1	1	1	0
21	1	1	1	0	1	1	1	0	0	0
22	1	1	1	1	1	1	1	1	0	0
23	1	1	1	1	1	1	1	1	0	0
24	0	1	1	1	0	1	1	1	1	0
25	1	1	1	1	1	1	0	1	0	0
26	1	1	1	1	1	1	0	1	0	0
27	1	1	1	1	1	1	1	1	0	0
28	1	1	1	1	1	1	1	0	1	0
29	1	1	1	1	1	1	1	1	0	0
30	0	1	1	1	1	1	1	1	0	0
31	1	1	1	1	1	1	0	0	0	0
32	1	1	1	1	1	1	1	1	0	0
33	0	1	1	1	1	1	1	1	0	0
34	1	1	0	1	1	1	1	1	1	0
35	1	1	0	1	1	1	1	0	1	0
36	0	1	1	0	0	1	1	0	0	0
37	1	1	1	1	0	1	1	0	1	0
38	1	1	0	1	1	1	1	1	0	0
39	1	1	1	1	1	0	1	1	0	0
40	1	1	1	1	1	1	1	1	1	0

Figure E.1: Observed Raw Data: Intel 3945

Appendix F: Performance of Intel 4965

	Intel 4965									
	Toshiba Satellite									
	1 denotes positive response; 0 denotes negative response									
	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
1	1	1	1	1	0	0	0	0	0	0
2	1	1	0	1	0	0	1	0	0	0
3	0	1	1	1	0	0	0	0	0	0
4	0	1	0	0	1	0	0	0	0	0
5	1	0	1	0	1	0	0	0	0	0
6	0	1	0	1	0	0	0	0	0	0
7	1	1	1	0	0	0	0	0	0	0
8	0	0	1	0	0	0	0	0	0	0
9	1	1	1	0	1	0	0	0	0	0
10	0	1	0	0	0	0	0	0	0	0
11	1	1	1	1	0	0	0	0	0	0
12	1	1	0	0	1	0	0	0	0	0
13	1	1	1	0	1	0	0	0	0	0
14	1	0	0	0	1	0	0	0	0	0
15	1	1	1	1	0	0	0	0	0	0
16	1	0	0	0	0	0	0	0	0	0
17	1	0	0	1	0	1	0	0	0	0
18	1	1	1	0	0	0	0	0	0	0
19	1	1	0	1	1	0	0	0	0	0
20	1	1	1	1	0	0	0	0	0	0
21	1	0	1	1	0	1	0	0	0	0
22	1	1	1	1	1	1	0	0	0	0
23	1	1	1	0	0	0	0	0	0	0
24	1	1	0	0	0	0	0	0	0	0
25	1	1	1	1	1	0	0	0	0	0
26	1	1	0	0	0	0	0	0	0	0
27	1	1	1	0	0	0	0	0	0	0
28	1	1	0	0	0	1	0	0	0	0
29	1	1	1	1	1	0	0	0	0	0
30	1	1	0	1	0	0	0	0	0	0
31	1	1	0	0	1	1	0	0	0	0
32	0	1	0	1	0	0	0	0	0	0
33	1	1	0	0	0	0	0	0	0	0
34	1	0	1	1	0	0	0	0	0	0
35	1	0	1	1	1	0	0	0	0	0
36	0	1	1	0	0	0	0	0	0	0
37	1	1	1	1	1	0	0	0	0	0
38	1	1	1	1	0	0	0	0	0	0
39	1	1	1	1	0	0	0	0	0	0
40	1	1	0	0	0	0	0	0	0	0

Figure F.1: Observed Raw Data: Intel 4965

Appendix G: Performance of Intel 6205

	Intel 6205									
	HP EliteBook 8570 Series									
	1 denotes positive response; 0 denotes negative response									
	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
1	0	1	1	1	0	1	1	0	0	0
2	0	0	1	1	1	1	0	0	0	0
3	1	1	1	1	1	0	0	0	0	0
4	1	1	1	1	1	1	0	0	0	0
5	1	1	1	1	0	1	0	0	0	0
6	1	1	1	1	0	0	0	0	0	0
7	1	0	1	1	0	1	0	0	0	0
8	1	0	1	1	1	0	0	0	0	0
9	0	1	1	1	1	1	0	0	0	0
10	1	1	1	1	1	0	0	0	0	0
11	0	1	1	1	1	1	0	0	0	0
12	0	1	1	1	0	1	0	0	0	0
13	1	0	1	1	1	1	0	0	0	0
14	1	1	1	1	1	1	0	0	0	0
15	1	1	1	1	1	1	0	0	0	0
16	1	1	1	1	0	1	0	0	0	0
17	1	1	1	0	1	1	0	0	0	0
18	1	0	1	1	1	1	0	0	0	0
19	0	1	1	1	1	1	0	0	0	0
20	1	1	1	1	1	0	0	0	0	0
21	1	0	1	1	1	1	1	0	0	0
22	1	1	1	1	0	0	0	0	0	0
23	1	1	1	1	0	1	0	0	0	0
24	1	1	0	0	1	1	0	0	0	0
25	0	1	1	1	1	1	1	0	0	0
26	1	1	1	1	0	1	0	0	0	0
27	1	1	1	1	1	0	0	0	0	0
28	1	1	1	1	1	1	0	0	0	0
29	1	1	1	1	1	1	0	0	0	0
30	1	1	1	1	1	0	0	0	0	0
31	1	1	1	1	1	0	0	0	0	0
32	1	1	1	1	1	0	0	0	0	0
33	1	1	1	1	1	1	0	0	0	0
34	0	1	0	1	1	1	0	0	0	0
35	0	1	1	1	0	1	0	0	0	0
36	1	1	1	1	1	0	0	0	0	0
37	1	1	1	0	1	1	0	0	0	0
38	1	0	0	1	1	1	0	0	0	0
39	1	1	1	1	0	1	0	0	0	0
40	1	1	1	1	0	0	0	0	0	0

Figure G.1: Observed Raw Data: Intel 6205

Appendix H: Performance of Intel 6250

	Intel 6250									
	Dell Latitude E6520 Series									
	1 denotes positive response; 0 denotes negative response									
	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
1	1	1	0	1	1	1	0	0	0	0
2	1	1	1	0	0	1	0	0	0	0
3	1	1	1	1	1	0	0	0	0	0
4	1	0	1	1	1	0	0	0	0	0
5	1	1	1	1	1	1	0	0	0	0
6	1	1	0	0	0	1	1	0	0	0
7	1	0	1	1	1	1	0	0	0	0
8	1	1	1	1	0	1	1	0	0	0
9	1	0	0	1	0	1	0	0	0	0
10	0	1	1	1	1	0	0	0	0	0
11	1	1	1	1	1	0	0	0	0	0
12	1	1	0	1	1	0	0	0	0	0
13	1	1	0	1	1	0	0	0	0	0
14	0	1	1	1	1	1	0	0	0	0
15	1	1	1	1	1	0	1	0	0	0
16	1	1	1	1	1	1	0	0	0	0
17	0	0	1	1	1	1	0	0	0	0
18	0	1	1	1	1	0	0	0	0	0
19	1	1	0	0	1	1	0	0	0	0
20	1	1	1	1	1	0	0	0	0	0
21	1	1	1	1	1	0	0	0	0	0
22	1	1	1	1	1	1	0	0	0	0
23	0	1	1	1	1	0	0	0	0	0
24	1	1	1	1	1	1	0	0	0	0
25	1	1	1	0	1	0	0	0	0	0
26	0	0	1	1	1	1	0	0	0	0
27	1	1	1	1	1	1	0	0	0	0
28	1	1	1	1	1	1	0	0	0	0
29	1	1	1	1	1	0	0	0	0	0
30	1	1	1	1	1	0	0	0	0	0
31	1	1	1	1	0	1	0	0	0	0
32	1	1	1	1	1	1	0	0	0	0
33	1	1	1	1	0	1	0	0	0	0
34	1	1	1	1	1	1	0	0	0	0
35	0	1	1	1	1	1	1	0	0	0
36	1	1	1	1	1	1	0	0	0	0
37	1	1	1	0	0	1	0	0	0	0
38	1	1	0	1	1	0	0	0	0	0
39	1	1	1	1	0	1	0	0	0	0
40	1	0	0	1	0	0	0	0	0	0

Figure H.1: Observed Raw Data: Intel 6250

Appendix I: MATLAB Monte Carlo Script

```
times_threshold_occurred = 0; % Counts how many times rand() is below threshold
for i = 1:35    % This was used for Trial 3, requiring 35 test packets to be sent
        x = rand();     % Generate a random value for x
        if x < .175     % Packet response rate of AR 9001-U 2NX during Trial 3
                times_threshold_occurred = times_threshold_occurred + 1;
        end
end

% The if statement can be structured to read "if x >= .875" as well
% Since the AR 9001-U 2NX had a low packet response rate, < .175
% intuitively is more clear
% 'i' is incremented up to the optimal number of test packets required to
% achieve > 99% accuracy, NOT up to the number of transmissions in each trial
```

Bibliography

[BhA12] Bhaya, Wesam S., and Samraa A. Al Asady. "Prevention of Spoofing Attacks in the Infrastructure Wireless Networks." 2012. Journal of Computer Science. Last Accessed: 15 Nov. 2013 <http://www.thescipub.com/abstract/10.3844/jcssp.2012.1769.1779>.

[Bur14] Burp Proxy. Last Accessed: 14 April 2014 < http://www.portswigger.net/Burp/proxy.html>

[BXY+06] Baxley, Thomas, Jinsheng Xu, Huiming Yu, Jinghua Zhang, Xiaohong Yuan, and Joseph Brickhouse. "LAN Attacker: A Visual Education Tool", *Proceedings of the 2006 Information Security Curriculum Development Conference*, pp. 137-142, 2006. Last Accessed: 24 Jan. 2014 < http://dl.acm.org/citation.cfm?id=1231072 >

[CBC06] Corbett, Cherita L., Raheem A. Beyah, and John A. Copeland. "Using Active Scanning to Identify Wireless NICs", *Proceedings of the 2006 IEEE Workshop on Information Assurance*. pp. 239-246, 2006. Last Accessed: 22 Aug. 2013 <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1652101>.

[CSS11] Chumchu, Prawit, Tanatat Saelim, and Chunyamon Sriklauy. "A New MAC Address Spoofing Detection Algorithm using PLCP Header" *Proceedings of International Conference on Information Networking*, pp. 48-53, 2011. Last Accessed: 18 Jan. 2014 < http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=05723112>.

[DAB+02] Doufexi, Angela, Simon Armour, Michael Butler, Andrew Nix, David Bull, and Joseph McGeehan. "A Comparison of the HIPERLAN/2 and IEEE 802.11a Wireless LAN Standards", *Communications Magazine, IEEE*, vol. 40, no. 5, pp. 172-180, 2002.

[Ell08] Ellingson, Steve. "Matrix Channel Measurement System". Last accessed: 26 Nov 2013 < http://www.ece.vt.edu/swe/mcms/>

[Erg02] Ergen, Mustafa. "IEEE 802.11 Tutorial", June 2002. University of California Berkeley. Last Accessed: 19 Nov. 2013 <http://wow.eecs.berkeley.edu/ergen/docs/ieee.pdf>.

[FlD01] Fleck, Bob and Jordan Dimov, "Wireless Access Points and ARP Poisoning", [Online Document] 12 Oct 2001. Last Accessed: 27 Jan. 2014 <http://bandwidthco.com/whitepapers/netforensics/arprarp/Wireless%20Access%20Points%20and%20ARP%20Poisoning.pdf>

[FLM10] Farooq, Taimur, David Llewellyn-Jones, and Madjid Merabti. "MAC Layer DoS Attacks in IEEE 802.11 Networks", *Proceedings of 11th Post-Graduate Symposium*, pp. 25-30, 2010 Last Accessed: 19 Nov. 2013 <http://www.cms.livjm.ac.uk/pgnet2010/MakeCD/Papers/2010063.pdf>.

[Gas05] Gast, Matthew. *802.11 Wireless Networks: The Definitive Guide*, 2nd Ed, Sebastopol, CA: O'Reilly Media, Inc, pp. 268-270, 2005.

[Gnu04] GNU Projects. "Macchanger". Last Accessed: 15 April 2014 < http://ftp.gnu.org/gnu/macchanger/>

[GuC06] Guo, Fanglu, and Tzi-cker Chiueh. “Sequence Number-Based MAC Address Spoof Detection”, *Proceedings of the 8th International Conference on Recent Advances in Intrusion Detection*, pp. 309-329, 2006. Last Accessed: 19 Nov. 2013 < http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.81.3346&rep=rep1&type=pdf>

[IEE14] IEEE. “Organizationally Unique Identifier”. Last Accessed: 14 April 2014 <http://standards.ieee.org/develop/regauth/oui/oui.txt>

[IEE99] IEEE. "Wireless LAN Medium Access Control and Physical Layer Specifications." 1999 ANSI/IEEE. Last Accessed: 13 Sept. 2013 <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=01389197>.

[KRM14] Kulesza, Nicholas J., Ben W. Ramsey, and Barry E. Mullins. “Wireless Intrusion Detection through Preamble Manipulation”, *Proceedings of the 2014 International Conference on Cyber Warfare and Security*, pp. 132-139, 2014.

[KRT02] Kridel, Don, Paul Rappoport, and Lester Taylor, *Forecasting the Internet: Understanding the Explosive Growth of Data Communications*, New York, NY: Springer Publisher, pp. 10-12, 2002

[KuR10] Kurose, James F., Keith W. Ross, *Computer Networking: A Top-Down Approach*, 5th Ed, Boston, MA: Addison-Wesley, pp. 14-15, 536-554, 2010.

[LSN02] Lansford, Jim, Adrian Stephens, and Ron Nevo. “Wi-Fi (802.11b) and Bluetooth: Enabling Coexistence”, *Network, IEEE*, vol. 15, no. 5, pp. 20-27, 2001.

[MEG08] Mahmood, Aneeq, Reinhard Exel, and Georg Gaderer. "Coherent Preamble Detection and Packet Decoding for Wireless Clock Synchronization using IEEE 802.11b WLAN", *Proceedings of the 2008 IEEE International Workshop on Factory Communication Systems*, pp. 105-108, 2008. Last Accessed: 22 Aug. 2013 <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=4638755&contentType=Conference+Publications>.

[MGS10] Misra, Sudip, Ashim Ghosh, and A.P. Sagar. “Detection of Identity-Based Attacks in Wireless Sensor Networks Using Signalprints”, *Proceedings of the 2010 IEEE/ACM International Conference on Green Computing and Communications*, pp. 35-41, 2010. Last Accessed: 20 Oct. 2013 <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5724808>.

[Min14] Minitab Software. “Minitab 17”. Last Accessed: 14 April 2014 < http://www.minitab.com/en-us/products/minitab/>

[Mol05] Molisch, Andreas F., *Wireless Local Area Networks*, 2nd Ed, Chichester, England: John Wiley & Sons, pp. 731-750, 2005.

[Nat12] National Instruments, “Record and Playback Demo with NI USRP”. [Online Document] 17 Apr. 2012. Last Accessed: 4 Jan. 2014 <http://www.ni.com/white-paper/13881/en/>.

[OHP05] O’Hara, Bob, Al Petrick, *IEEE 802.11 Handbook: A Designer’s Companion*, 2nd Ed, Chichester, England: IEEE-Press, pp. 13-15, 2005.

[Pon12] "2012 Cost of Cyber Crime Studies: United States." Oct. 2012. Ponemon Institute LLC. Last Accessed: 25 Sept. 2013 <http://www.ponemon.org/local/upload/file/2012_US_Cost_of_Cyber_Crime_Study_FINAL6%20.pdf>.

[RMT13] Ramsey, Benjamin W., Barry E. Mullins, and Michael A. Temple. *Wireless Intrusion Detection and Device Fingerprinting through Preamble Manipulation*. 22 Aug. 2013.

[RTM10] Reising, Donald R., Michael A. Temple, and Michael J. Mendenhall. "Improving Intra-Cellular Security Using Air Monitoring with RF Fingerprints", *Proceedings of the 2010 Wireless Communications and Networking Conference*. pp. 1-6, 2010. Last Accessed: 11 Feb. 2014 <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4698196>.

[STC08] Sheng, Yong, Keren Tan, and Guanling Chen. "Detecting 802.11 MAC Layer Spoofing Using Received Signal Strength", *Proceedings of the 27th Conference on Computer Communications*. 2008. Last Accessed: 10 Oct. 2013 <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4509834>.

[STM08] Suski, William C., Michael A. Temple, and Michael J. Mendenhall. "Using Spectral Fingerprints to Improve Wireless Network Security" *Proceedings of Global Telecommunications Conference*, pp. 1-5, 2008. Last Accessed: 23 Sept. 2013 <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4698196&tag=1>.

[TVP10] Trostle, Jonathan, Bill Van Besien, and Ashish Purjari. "Protecting Against DNS Cache Poisoning Attacks". *Proceedings of the 2010 6th IEEE Workshop on Secure Network Protocols (NPSec)*, pp. 25-30, 2010. Last Accessed: 11 Dec. 2013 < http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5634454>

[Voc12] "802.11a White Paper" VOCAL Technologies, Ltd. Last Accessed: 24 June 2013 <http://www.vocal.com/wp-content/uploads/2012/05/80211a_wp1pdf.pdf>.

[Wri03] Wright, Joshua. "Detecting Wireless LAN MAC Address Spoofing." 21 Jan. 2003. Last Accessed: 22 Aug. 2013 <http://www.willhackforsushi.com/papers/wlan-mac-spoof.pdf>.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
30–06–2014	Master's Thesis	Aug 2012–June 2014

4. TITLE AND SUBTITLE

RADIO FREQUENCY FINGERPRINTING TECHNIQUES THROUGH PREAMBLE MODIFICATION IN IEEE 802.11b

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Kulesza, Nicholas J., Captain, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB, OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT–ENG–T–14–J–8

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

INTENTIONALLY LEFT BLANK

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Wireless local area networks are particularly vulnerable to cyber attacks due to their contested transmission medium. Access point spoofing, route poisoning, and cryptographic attacks are some of the many mature threats faced by wireless networks. Recent work investigates physical-layer features such as received signal strength or radio frequency fingerprinting to identify and localize malicious devices. This thesis demonstrates a novel and complementary approach to exploiting physical-layer differences among wireless devices that is more energy efficient and invariant with respect to the environment than traditional fingerprinting techniques. Specifically, this methodology exploits subtle design differences among different transceiver hardware types.

A software defined radio captures packets with standard-length IEEE 802.11b preambles, manipulates the recorded preambles by shortening their length, then replays the altered packets toward the transceivers under test. Wireless transceivers vary in their ability to receive packets with preambles shorter than the standard. By analyzing differences in packet reception with respect to preamble length, this methodology distinguishes amongst eight transceiver types from three manufacturers. All tests to successfully enumerate the transceivers achieve accuracy rates greater than 99%, while transmitting less than 60 test packets.

This research extends previous work illustrating RF fingerprinting techniques through IEEE 802.15.4 wireless protocols. The results demonstrate that preamble manipulation is effective for multi-factor device authentication, network intrusion detection, and remote transceiver type fingerprinting in IEEE 802.11b.

15. SUBJECT TERMS

Wireless, Preamble, 802.11b, RF, Fingerprinting

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Barry E. Mullins (ENG)
U	U	U	UU	100	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636 x7979; barry.mullins@afit.edu